

Habilitation à diriger les Recherches
en
INFORMATIQUE

présentée et soutenue publiquement le 5 décembre 2007 par
DAVID COEURJOLLY

**Algorithmique pour l'analyse et la
modélisation en géométrie discrète**

Laboratoire LIRIS - UMR 5205 CNRS
Ecole Doctorale “Informatique et Information pour la Société”

COMPOSITION DU JURY

Mme.	BORGEFORS Gunilla	Rapporteur	Professeur, CBA, Uppsala Universitet, Suède
M.	BRAQUELAIRE Achille	Rapporteur	Professeur, LaBRI, Université Bordeaux 1
M.	MAÎTRE Henri	Rapporteur	Professeur, TSI, ENST, Paris
Mme	MONTANVERT Annick	Examinateur	Professeur, GIPSA-Lab, UPMF, Grenoble
M.	DEVILLERS Olivier	Examinateur	Directeur de Recherches, INRIA, Sophia-Antipolis
M.	PÉROCHE Bernard	Examinateur	Professeur, LIRIS, Université Lyon 1

Remerciements

Je tiens à remercier les membres de mon jury qui m'ont fait l'honneur et le plaisir de participer à cette habilitation. Tout d'abord, merci à GUNILLA BORGEFORS d'avoir accepté de rapporter sur ce manuscrit en français et qui a amélioré considérablement mon suédois en faisant passer mon vocabulaire de 1 à 5 mots.

Merci à ACHILLE BRAQUELAIRE d'avoir accepté (une nouvelle fois) de rapporter sur mon manuscrit. Tout comme à HENRI MAÎTRE qui a eu la difficulté de rapporter un travail sans doute éloigné de ses propres activités. Ses remarques et conseils lors des échanges que nous avons pu avoir ont été cependant très riches.

Un grand merci à ANNICK MONTANVERT à qui je dois beaucoup, tant d'un point de vue scientifique que d'un point de vue personnel.

Merci à OLIVIER DEVILLERS pour sa participation à ce jury un peu particulier, même si finalement tant de choses sont communes entre géométrie algorithmique et géométrie discrète. Merci enfin à BERNARD PÉROCHE pour ses conseils en tant que directeur et collègue.

Je tiens aussi à remercier toutes les personnes que j'ai rencontrées en conférences ou en séminaires. De nombreux développements sont nés de ces échanges ¹

Je terminerai par tous mes collègues lyonnais, anciens de Lyon 2 ou actuels (du LIRIS et de M2DisCo) qui seront bien déçus de voir le couard que je suis de ne pas me lancer dans un listage nominatif aux risques d'en oublier. Les concernés se reconnaîtront sans problème.

Last but not least, merci à ma famille, à Anne et à Lisa pour leur support moral, affectif et parfois bruyant.

1. ok je vous avouerai que ces deux dernières phrases étaient déjà dans les remerciements de ma thèse (et oui je suis en plein copier/coller ;)). Il n'en est pas moins vrai que ces dernières sont toujours autant d'actualité.

Table des matières

1 Rapport d'activités	3
1.1 Notice Individuelle	4
1.2 Emplois et formation	4
1.3 Activités d'enseignement	5
1.4 Activités de recherche	6
1.5 Encadrement	9
1.6 Activités contractuelles	10
1.7 Activités administratives et collectives	10
1.8 Publications	11
2 Introduction générale	13
3 Notions de base	17
3.1 Image, image numérique et support d'image	18
3.2 Espace discret et connexité	18
3.3 Objets, courbes, surfaces	23
3.4 Eléments géométriques	25
3.5 Eléments de complexité	28
4 Modèle analytique discret, arithmétique d'intervalles et supercouverture	33
4.1 Introduction	34
4.2 Modèles de discréétisation	34
4.3 Arithmétique d'intervalles et supercouverture	38
4.4 Conclusion	45
5 Algorithmes de reconnaissance de parties linéaires et de cercles	47
5.1 Introduction	48
5.2 Définitions des objets élémentaires	48
5.3 La boîte à outils	51
5.4 Algorithmes de reconnaissance	55
5.5 Conclusion	64
6 Reconstruction réversible d'objets discrets par des parties linéaires	65
6.1 Introduction	66
6.2 Reconstruction 2D sur grille isothétique irrégulière	66

TABLE DES MATIÈRES

6.3	Reconstruction réversible en dimension 3	73
6.4	Conclusion	77
7	Algorithmes séparables pour l'analyse volumique de formes	83
7.1	Introduction	84
7.2	État de l'art	84
7.3	Transformation en distance, reconstruction et axe médian discret	85
7.4	Généralisation aux espaces toriques	101
7.5	Conclusion	104
8	Minimalité et simplification de l'axe médian	107
8.1	Introduction	108
8.2	L'axe médian minimum est NP-complet	108
8.3	Simplification réversible de l'axe médian	113
8.4	Filtrage de l'axe médian et liens avec la géométrie algorithmique	114
8.5	Application : compression sans perte et transmission progressive	116
8.6	Conclusion et perspectives	120
	Conclusions et perspectives	125
8.7	Synthèse des travaux réalisés	125
8.8	Perspectives et projets	126
A	Réalisations logicielles et diffusion	129

Table des figures

3.1	Détail d'une mosaïque à Herculaneum, représentant Amphitrite.	19
3.2	Représentation d'une image sur un support discret régulier.	19
3.3	Pavages (et maillages) réguliers en dimension deux par cellules carrées, triangulaires et hexagonales.	20
3.4	Représentation et codage des pavages réguliers 2D sous forme matricielle. . .	20
3.5	Structuration possible de quelques grilles isothétiques irrégulières.	21
3.6	En 3D, différents voisinages du cube central avec le pavage régulier.	22
3.7	A gauche, A et B sont (1)- et (0)-adjacents, à droite, A et B sont (0)-adjacents mais pas (1)-adjacents.	22
3.8	Illustration d'un remplissage d'objet à partir du pixel noir en (1)-connexité (<i>gauche</i>) et en (0)-connexité (<i>droite</i>). Les flèches illustrent un exemple de propagation selon la connexité considérée.	24
3.9	Décomposition d'un pixel et d'un voxel en cellules de dimensions inférieures. .	24
3.10	Exemples d'objets topologiques sur une grille isothétique irrégulière : (<i>de gauche à droite</i>) un (1)-objet (mais non (0)-objet), une (0)-courbe et une (0)-courbe fermée.	24
3.11	Seule la courbe (a) est un segment discret selon les définitions usuelles des droites discrètes.	25
3.12	Exemple de reconstruction réversible en dimension 2 et 3.	26
3.13	Illustration de la représentation d'une forme par union de disques. Les centres des disques (pointillés) représentent schématiquement l'axe médian de la forme. .	27
3.14	Illustration de distances discrètes entre deux points discrets. Les chemins correspondent à des plus courts chemins pour la distance considérée (les déplacements élémentaires et leurs poids sont indiqués par les flèches).	28
3.15	Représentation planaire de l'instance de PLANAR-3-SAT : $\phi = (x_1 \vee \neg x_3 \vee x_4) \wedge (\neg x_1 \vee \neg x_2 \vee x_5) \wedge (x_1 \vee x_3 \vee \neg x_4)$. Les clauses c_1, c_2 et c_3 sont numérotées dans l'ordre de la formule et les deux types de liens correspondent à l'utilisation de la variable de manière négative ou positive.	30
4.1	Rôle de la résolution pour une cohérence entre la topologie de l'objet continu par rapport à l'objet discret, avec une discréttisation en supercouverture (voir paragraphe 4.2.3).	34
4.2	Processus de discréttisation classiques sur le maillage (première ligne) ou sur le pavage (seconde ligne) : à gauche, discréttisation de GAUSS d'un objet et à droite, discréttisation GIQ d'une courbe.	35
4.3	Différents schémas de discréttisation spécifiques aux droites : (<i>de gauche à droite</i>) discréttisations GIQ, OBQ et BBQ. Les intervalles représentent les opérations d'arrondi.	36

TABLE DES FIGURES

4.4	Processus de discréétisation classiques sur le pavage (première ligne) ou sur le maillage (seconde ligne) : (<i>de gauche à droite</i>) discréétisation en supercouverture d'une courbe, discréétisation en supercouverture d'un objet, illustration d'une bulle pour la discréétisation d'un segment de droite, et discréétisation en modèle standard.	37
4.5	Quelques exemples de modèles analytiques : supercouverture, naïf fermé, pythagoricien et engendré par un pentagone.	37
4.6	Illustration des définitions d'un modèle analytique d'une forme quelconque : définitions par intersection avec les régions et par morphologie mathématique (C est la région et M sa réflexion par rapport au centre de C).	38
4.7	Analyse par arithmétique d'intervalles $\mathbb{I}_{\mathbb{F}}$ de la fonction $f(x) = (x+1)(x-1)$ sur $[-1, 1]$ par des intervalles $\{X_i\}_{i=1 \dots 20}$ de taille $1/10$	40
4.8	Tracé en arithmétique d'intervalles $\mathbb{I}_{\mathbb{Z} + \frac{1}{2}}$ des fonctions $f(x) = \frac{1}{2}x + 1$ (a) et $f(x) = \frac{1}{3}x$ (b).	43
4.9	Tracé en arithmétique d'intervalles $\mathbb{I}_{\mathbb{Z} + \frac{1}{2}}$ modifiée, des fonctions $f(x) = \frac{1}{3}x$ et $f(x) = 10 \sin(x) \exp^{-x^2/50}$	43
4.10	Exemple de tracé avec des extensions non optimales : (a) $f(x) = \frac{x^2}{x}$ et (b) $g(x) = h(h(x))$ avec $h(x) = \frac{\sqrt{x^2 - x + 1/2}}{\sqrt{x^2 + 1/2}}$	44
4.11	Illustration de la preuve du lemme 4.1 : les différents cas de position de C dans Y	44
5.1	Connexité des droites discrètes en fonction de ω	49
5.2	Préimage d'un ensemble de points discrets S	49
5.3	Morceau de plan discret naïf de paramètres $P(6, 13, 17, 17)$ (a) et illustration de la base 2D associé à un morceau de plan.	51
5.4	(a) Exemple de polygone convexe discret inclus. Les figures (b) et (c) illustrent la combinatoire entre l'objet discret (ici 3D) et les sommets de son enveloppe convexe (c).	53
5.5	Illustration des approches géométriques pour la reconnaissance d'hyperplans discrets : l'ensemble S initial, calcul d'un plan support, test de séparabilité S/S' et calcul de l'épaisseur géométrique de l'ensemble corde de S	56
5.6	Illustration de deux morceaux de plans discrets selon les générateurs proposés : (a) un morceau ayant une base rectangulaire (b) son enveloppe convexe et (c) sa préimage; (d) un morceau de plan à base "co-sphérique", (e) son enveloppe convexe et (f) sa préimage.	59
5.7	Analyse expérimentale du nombre de sommets de $CH(S)$ pour les générateurs à base hyper-rectangulaire (a) et co-sphérique (b) pour $n = \{3, 4, 5, 6\}$	60
5.8	Analyse expérimentale du nombre de sommets de $\mathcal{P}(S)$ pour les générateurs à base hyper-rectangulaire (a) et co-sphérique (b) pour $n = \{3, 4, 5, 6\}$	60
5.9	Analyse comparée de $ CH(S) $ et $ \mathcal{P}(S) $ pour la dimension 3 pour les plans discrets à base hyper-rectangulaire (a) et co-sphérique (b).	61
5.10	Illustration du processus global de filtrage pour la reconnaissance de disques discrets.	62
5.11	Illustration du filtrage de T : test d'inclusion dans un triangle (a) et décomposition du test, arête par arête (b).	62
5.12	Illustration de la définition des points de BEZOUT associés à un segment discret.	63

6.1	Évaluation de la fonction $f(x, y) = y^2 - x^3 + x - \frac{1}{2}$ sur l'intervalle $[-4, 4] \times [-4, 4]$ avec comme critère d'arrêt, une borne sur la largeur des intervalles fixée à : (b) 1.0, (c) 0.5 et (d) 0.1.	67
6.2	Illustration de la reconstruction par une courbe polygonale d'une (1)-courbe par la propagation d'un cône de visibilité.	67
6.3	Représentation classique du graphe de REEB en géométrie différentielle basée sur les points critiques de MORSE (a). Illustration du recodage d'un (1)-objet en (1)-courbe en fonction d'une fonction de hauteur h	68
6.4	(a) Exemple de construction parallèle du graphe de REEB et du recodage des cellules pour former des (1)-courbes pour les arcs du graphe. (b) Illustration complète avec reconstruction polygonale.	68
6.5	Tracés de la fonction $x^2 + y^2 + \cos(2\pi x) + \sin(2\pi y) + \sin(2\pi x^2) \cos(2\pi y^2) = 1$ sur $[-1, 1]^2$ (exemple de [Sny92]) : (a) approximation du tracé, (b) cellules résultantes de l'approximation en intervalles et du recodage, (c) le graphe de REEB associé. Les figures (d) à (g) illustrent les étapes de la reconstruction polygonale partant des noeuds du graphe de REEB.	69
6.6	Illustration complète pour la fonction $f(x, y) = y^2 - x^3 + x - \frac{1}{2}$ sur l'intervalle $[-4, 4] \times [-4, 4]$ (\mathcal{C} , voir figure 6.1). \mathcal{L} correspond aux reconstructions polygonales et \mathcal{G} , au graphe de REEB associé.	70
6.7	Tracés de la fonction présente dans la figure 6.5 avec raffinements successifs guidés par l'arithmétique (deux premières lignes) et fusions successives par interactions (deux lignes suivantes). Les zones en bleu correspondent aux zones associées aux modifications.	71
6.8	Reconstruction sur un objet (1)-connexe classique qui n'est pas une simple (1)-courbe.	71
6.9	Exemple de reconstruction d'un objet issu d'un document numérisé : (a) image initiale, (b) graphe de REEB représentant la forme et (c) la reconstruction.	72
6.10	Les 14 configurations de l'algorithme initial des <i>Marching-Cubes</i>	73
6.11	Objet discret 3D et une surface MC associée.	74
6.12	Illustration du transfert des étiquettes des sommets vers les faces. Les triangles avec des croix sont 2-NH ou 3-NH.	74
6.13	Illustration du processus de simplification des triangles homogènes : (de gauche à droite) un objet discret, une surface des MC, segmentation de la surface en morceaux de plans (étiquettes ramenées aux triangles) et simplification des ensembles connexes de triangles homogènes.	75
6.14	(gauche) : Deux reconstructions possibles validant la propriété de réversibilité. (droite) : nous restreignons la reconstruction à être contenue dans la zone grise.	79
6.15	Illustration de la suppression d'une arête et de la suppression d'un triangle.	79
6.16	Décomposition bidimensionnelle d'une cellule de MC et faces associées à un triangle.	80
6.17	Illustration du traitement en dimension 2 de la suppression d'un triangle T sur les faces de la cellule du MC associée.	80
6.18	Quelques résultats de la simplification : (de gauche à droite) les objets discrets, la surface MC, première simplification des triangles homogènes uniquement et enfin, simplification finale H+NH.	81
7.1	Illustration du calcul de la transformée en distance de manière séparable (dimension par dimension).	86

TABLE DES FIGURES

7.2 Illustration du calcul de l'enveloppe inférieure de paraboles : si [16, 1, 4, 1] est une colonne de G après l'étape 1 (5 ^{ème} colonne dans la figure 7.1, valeurs élevées au carré), on obtient (à gauche) l'ensemble des paraboles $g(i, y)^2 + (j - y)^2$ et (à droite) la courbe épaisse correspondant à l'enveloppe inférieure ; le résultat est [2, 1, 2, 1].	86
7.3 Algorithmes pour le calcul de la transformée en distance euclidienne d'un objet discret en dimension 2.	87
7.4 Exemples de transformées en distance en dimension 2 avec différentes représentations : objets binaires, valeur de la transformée en distance, représentation des valeurs sur une plage réduite de niveaux de gris et représentation surfacique (la hauteur correspond à la valeur de la transformée).	88
7.5 Représentation d'une transformée en distance euclidienne 3D par coupe dans le volume (ici avec une palette circulaire de niveaux de gris).	88
7.6 Représentation de la décomposition en dimension pour le calcul du diagramme de VORONOÏ discret. (a) : Image en entrée (les pixels de \bar{X} sont en gris), et résultat de l'étiquetage selon l'axe des x . Les collisions sont indiquées par des doubles lettres. (b) : la valeur de transformée en distance de la colonne $i = 4$. (c) : L'ensemble des paraboles pour la colonne $i = 4$. Les points indiquent les collisions ($\#\Pi_{Sp} > 1$) détectées dans la dimension précédente. (d) : l'enveloppe inférieure des paraboles et détection des nouveaux points tels que $\#\Pi_{Sp} > 1$: le point $j = 7$ n'est plus sur une arête du diagramme, le point $j = 1$ est préservé et un nouveau point équidistant+ de plusieurs sites a été détecté $j = 4$	90
7.7 Résultats obtenus par l'algorithme de calcul du diagramme de VORONOÏ discret sur l'objet de la figure 7.6 (a – c) et sur un rectangle désaxé (d – f). Nous traçons, l'étiquetage sous forme de graphe orienté de $\Pi_{\bar{X}}$ (a) et (e), d'étiquetage de VORONOÏ (affectation arbitraire sur les bords, (b) et (e)) et points discrets pour lesquels $\Pi_{\bar{X}}(p) > 1$ (c) et (f).	92
7.8 Résultats sur un disque discret.	93
7.9 Tracés de k en fonction de m (axe horizontal) : (a – b) formes 2D (rectangles "r", ellipses "e", images quelconques "m"), et (c) formes 3D (parallélépipèdes rectangles "b", tores "t" et images de vertèbres 3D "v") [CCZ07].	93
7.10 Distribution de $ \Pi_{\bar{X}}(p) $ sur une forme 2D (a) et sur une vertèbre 3D (b) [CCZ07].	94
7.11 Illustration du processus dimension par dimension pour la reconstruction en distance euclidienne.	95
7.12 Illustration du processus dimension par dimension pour la reconstruction en distance euclidienne.	96
7.13 Illustration de la différence entre axe médian et squelette dans le cas continu : l'arête fermée appartient à l'axe médian mais non au squelette qui ne contient que l'arête ouverte.	97
7.14 Boules incluses dans une forme et dômes associés.	97
7.15 De gauche à droite, une forme discrète X et sa transformée en distance euclidienne ; les 5 centres des boules dont le dôme touche l'enveloppe supérieure des paraboloïdes, dessinée à droite ; l'axe médian discret est formé du seul point central encerclé ; forme continue Y et enveloppe supérieure des paraboloïdes bordant la forme \hat{Y}	98

7.16	Une illustration de la différence entre l'ensemble $AM_0(Y)$ et l'axe médian discret. A gauche : les segments en pointillé représentent les boules euclidiennes 1D associées à $AM_0(Y) = \{\hat{D}, \hat{E}\}$ alors que les segments en traits pleins représentent les boules discrètes 1D. A droite : illustration du processus de filtrage permettant de ne conserver que les boules discrètes 1D maximales.	99
7.17	Extraction de l'AMD en dimension 2 : la première ligne présente les objets discrets, la seconde présente les centres de boules du AMD (pixels en noir).	99
7.18	AMD en dimension 3 : la première ligne présente les objets discrets, la seconde ligne présente les centres des boules de l'AMD.	100
7.19	(a) Diagramme de puissance associé à deux boules de rayons différents. (b) Représentation surfacique du diagramme avec des paraboloïdes elliptiques.	101
7.20	(a) et (b) construction d'un pavage de l'espace par la répétition régulière d'un motif (a), (c) représentation de l'espace torique et (d) exemple de disque "torique".	102
7.21	Illustration de la première étape pour la transformée en distance torique : sélection du point de rupture, recodage en vecteur et réécriture du résultat.	102
7.22	Illustration du recodage de la liste circulaire [1, 2, 4, 5, 1, 2] pour le calcul de transformation inverse ou l'extraction de l'axe médian. Le point $i = 3$ est utilisé comme point de rupture.	103
7.23	Étapes de la transformation en distance sur une grille torique.	103
7.24	Résultat des algorithmes sur le complémentaire du motif de la figure 7.20-(d). (a) Transformée torique du motif et (b) illustration du pavage de l'espace par translation régulière du résultat. (c) Extraction de l'axe médian torique de l'image (a) et (d) la répétition de l'axe médian.	104
7.25	Représentation de transformées en distance toriques : (a) transformée sur une répétition d'un motif défini comme étant le complémentaire d'un cube, (b) coupe dans le volume de l'image (a). (c - d) Autre exemple à partir d'un motif défini par le complémentaire de l'objet A1.	105
7.26	Reconstruction sur un espace torique 3D pour générer un texture régulière 3D de sphères : (a) motif avec un placement aléatoire des centres et des valeurs des rayons, (b) transformation inverse et (c) texture obtenue par translation du motif (les sphères discrètes sont représentées par des sphères euclidiennes).	105
8.1	Axe médian discret pour la distance euclidienne (à gauche) et sous-ensemble de l'axe médian permettant une reconstruction exacte (à droite).	108
8.2	Schéma général de la réduction : graphe planaire instance de PLANAR-4-3-SAT, plongement orthogonal [Tam87] du graphe planaire et construction de l'objet discret associé à l'instance.	109
8.3	Schéma géométrique et couvertures minimales d'un littéral. Le schéma du haut correspond à l'assignation à Vrai de la variable.	110
8.4	Éléments de construction pour définir des liens.	111
8.5	Schéma géométrique pour coder les clauses.	112
8.6	Expérimentations de différents algorithmes de simplification d'axe médian : (<i>de gauche à droite</i>) les objets discrets, l'axe médian discret, la simplification de [RB93] et l'algorithme glouton. Le nombre de boules et le taux de simplification sont donnés sous les objets. Le temps de calcul est indiqué entre crochets.	114
8.7	Paramètres associés à un point s de l'axe médian : l'étanchéité $\rho(s)$ et l'angle bissecteur $\alpha(s) = \widehat{p_0 s p_1}$	115
8.8	Processus de filtrage en dimension 2	116

TABLE DES FIGURES

8.9 Illustration du filtrage en dimension 3 : graphes de filtrage et reconstruction du plus grossier au plus fin des objets discrets.	118
8.10 Illustration des primitives basées sur des interpolations de boules en dimension 2 et 3 et validation d'un tronc de cône par un test d'inclusion dans la forme.	118
8.11 Pseudo-tronc de cône et pseudo-triangle 3D.	119
8.12 Construction d'un pseudo-triangle 2D et d'un pseudo-tétraèdre 3D en utilisant les pseudo-troncs de cônes et les pseudo-triangles 3D.	120
8.13 Courbes débit/distorsion sur les objets <code>catenoid</code> de taille 8 (<i>a</i>) et <code>32Dodge</code> (<i>b</i>) avec plusieurs stratégies.	122
8.14 Reconstitution de l'objet <code>dodge</code> après 1kb, 2kb, 5kb, 10kb, et complète.	123

Liste des tableaux

3.1	Caractérisation des adjacences en dimension trois.	21
3.2	Lien entre les notations de voisinage pour un pavage par hypercubes.	22
5.1	Enveloppes convexes et principales références.	54
5.2	Principales références pour la séparabilité de deux ensembles par un cercle.	55
5.3	Algorithmes de reconnaissance de droite discrète.	55
5.4	Solutions algorithmiques pour la reconnaissance de plans et d'hyperplans discrets : m est le nombre de points discrets, n la dimension, E est la taille de la préimage, et D une borne sur la taille du domaine contenant l'ensemble S	58
5.5	Algorithme générique pour la reconnaissance de disques discrets et analyses de complexités pour les différentes options.	63
6.1	Quelques résultats quantitatifs sur les exemples de la figure 6.18.	77
8.1	Résultats quantitatifs des filtrages sur les objets des figures 8.8 et 8.9. Notez que si les seuils sont de $(0, 0)$, le résultat est exactement l'axe médian discret.	117

LISTE DES TABLEAUX

Chapitre 1

Rapport d'activités

Sommaire

1.1	Notice Individuelle	4
1.2	Emplois et formation	4
1.3	Activités d'enseignement	5
1.3.1	Résumé	5
1.3.2	Détails des enseignements	5
1.3.2.1	Responsabilités et vacations en temps que Chargé de Recherche	5
1.3.2.2	Enseignement en tant qu'Attaché Temporaire d'Enseignement et de Recherche	6
1.3.2.3	Enseignement en tant que Moniteur	6
1.4	Activités de recherche	6
1.4.1	Environnement et orientation générale	6
1.4.2	Collaborations internationales	7
1.4.3	Collaborations nationales extérieures au LIRIS	7
1.4.4	Animation de la recherche	7
1.4.4.1	IAPR Technical Committee “Discrete Geometry”	7
1.4.4.2	14th IAPR International Conference on Discrete Geometry for Computer Imagery	8
1.4.4.3	Organisation de manifestations nationales	8
1.4.4.4	Participation à des comités de programme et de lecture	8
1.4.4.5	Participation à des jurys de thèse	8
1.5	Encadrement	9
1.5.1	Encadrement d'étudiants en thèse	9
1.5.2	Encadrements de stagiaires	9
1.6	Activités contractuelles	10
1.7	Activités administratives et collectives	10
1.7.1	Au sein de l'Université Claude Bernard Lyon 1	10
1.7.2	Au sein du laboratoire	11
1.7.3	Au sein d'autres établissements	11
1.8	Publications	11
1.8.1	Ouvrages	11
1.8.2	Chapitre dans ouvrage	11
1.8.3	Revues internationales avec comité de lecture	11
1.8.4	Conférences internationales	11
1.8.5	Conférences nationales	11

1.1 Notice Individuelle

- ◊ *Nom* David COEURJOLLY
- Date et lieu de naissance* 3 septembre 1977 à Lille (59)
- Situation actuelle* Chargé de Recherche CNRS
- ◊ *Adresse professionnelle* Laboratoire LIRIS UMR 5205, Université Claude Bernard Lyon 1
 43 Bd du 11 novembre 1918
 Bat. Nautibus,
 69622 VILLEURBANNE
- Téléphone* 04 78 44 82 40
- Télécopie* 04 78 43 15 36
- ◊ *Mél* david.coeurjolly@liris.cnrs.fr
- Page web* <http://liris.cnrs.fr/~dcoeurjo>

1.2 Emplois et formation

- ◊ Depuis Décembre 2003 CHARGÉ DE RECHERCHE CNRS, Laboratoire LIRIS, UMR 5205.
 - ◊ Septembre 2003 - Décembre 2003 ATTACHÉ TEMPORAIRE D'ENSEIGNEMENT ET DE RECHERCHE à l'Institut National des Sciences Appliquées de Lyon (INSA), département Informatique, laboratoire de rattachement LIRIS UMR 5205.
 - ◊ Mars 2003 - Juin 2003 POST DOCTORAT, Laboratoire LIS, Université Joseph Fourier, Grenoble

<i>Sujet</i>	Problème du plus grand convexe inclus dans un polygone simple
<i>Direction</i>	JEAN-MARC CHASSERY
 - ◊ Sept. 2000 - Dec. 2002 DOCTORAT D'UNIVERSITÉ, Spécialité Informatique

<i>Lieu</i>	Université Lumière Lyon 2 – Laboratoire ERIC	
<i>Titre de la thèse</i>	Algorithmique et géométrie discrète pour la caractérisation des courbes et des surfaces	
<i>Directeurs</i>	Serge MIGUET et Laure TOUGNE	
<i>Soutenue</i>	le 18 décembre 2002	
<i>Jury</i>		
<i>rapporteur</i>	Jean-Pierre BRAQUELAIRE	Professeur
<i>rapporteur</i>	Jean-Marc CHASSERY	Directeur de Recherche CNRS
	Jean-Michel JOLION	Professeur
	Jean-Pierre REVEILLÈS	Professeur
<i>directeur</i>	Serge MIGUET	Professeur
<i>directeur</i>	Laure TOUGNE	Maître de conférences

1.3 Activités d'enseignement

1.3.1 Résumé

Filière	Matière	Type	Vol.
06-07	Master Recherche Informatique 2ième année, UCBL Lyon 1	Cours de spécialité recherche	TD 6h
05-06	Master Recherche Informatique 2ième année, UCBL Lyon 1 3ième année école d'ingénieurs IEG, Grenoble	Cours de spécialité recherche	CM 10h
		Cours de spécialité recherche	CM 8h
04-05	Master Recherche Informatique 2ième année, UCBL Lyon 1 3ième année école d'ingénieurs IEG, Grenoble	Cours de spécialité recherche	CM 10h
		Cours de spécialité recherche	CM 8h
03-03	ATER (demi-poste) département informatique, INSA, Lyon	Programmation C++	TP 80h
00-03	Moniteur à l'Institut de la Communication, Lyon2	-	TD 192h

1.3.2 Détails des enseignements

1.3.2.1 Responsabilités et vacations en temps que Chargé de Recherche

Master Recherche Université Claude Bernard Lyon 1 : cours de spécialité recherche (Master 2ième année) “Géométrie algorithmique et géométrie discrète” en collaboration avec RAPHAELLE CHAINE (MCF - UCBL). L’objectif de ce cours est d’introduire les fondements géométriques et topologiques permettant de formaliser des notions de formes discrètes, continues par morceaux (maillage, triangulation) ou induites par l’imagerie numérique (objets sur des grilles). Ce cours de spécialité se trouve dans le parcours “Informatique Graphique et Image” du Master Recherche Informatique de Lyon.

Master 1ère année Informatique Université Claude Bernard Lyon 1 : organisation du module de TER (Travaux d’Etude et de Recherche). L’objectif du TER est de confronter les étudiants au monde de la recherche après une première expérience du monde de l’entreprise en fin de licence, ceci afin de vous permettre un choix éclairé entre M2 recherche et M2 professionnel à l’issue du M1. En 2006-2007, cette organisation était en collaboration avec STÉPHANIE JEAN-DAUBIAS (MCF, UCBL). Pour l’année 2007-2008, j’ai la responsabilité seul de ce module.

Master recherche et 3ième année école à l’École Nationale Supérieure d’Ingénieurs Electriciens de Grenoble : cours commun entre la troisième année de la filière ATIS (analyse et traitement des images et des signaux) de l’IEG et le Master Recherche “Électronique, Electrotechnique, Automatique et Traitement du signal”, filière “Signal, Image, Parole, Télécoms” (SIPT). En collaboration avec CÉDRIC GÉROT (MCF - Université Pierre Mendès-France, Grenoble) et JOCELYN CHANUSSET (MCF, ENSIEG, Grenoble), l’objectif est d’introduire le vocabulaire et les notions de base pour l’analyse d’images (notions de grille, de topologie, de géométrie discrète) dans une problématique de segmentation de formes.

1.3.2.2 Enseignement en tant qu'Attaché Temporaire d'Enseignement et de Recherche

Au cours de ma période d'ATER à l'INSA de Lyon, j'ai participé au cours : *Algorithmique, programmation objet - C++* avec MATHIEU MARANZANA (MCF - INSA de Lyon). L'objectif de cet enseignement était pour les étudiants d'acquérir les compétences en développement de logiciels. Pour cela, nous insistions sur la maîtrise des outils méthodologiques indispensables à la conception, la réalisation, la validation et la maintenance de logiciels de qualité, illustrée avec le langage C++. Cet enseignement intervenait au début de la 3^e année de l'INSA (département Informatique) et a consisté en des séances de TP.

1.3.2.3 Enseignement en tant que Moniteur

A partir de l'année universitaire 2000-2001, mon enseignement s'est déroulé dans le cadre d'un monitorat d'Initiation à l'Enseignement Supérieur. Ce poste était rattaché à l'Institut de la Communication (ICOM) de l'université Lumière Lyon 2.

L'Institut de la Communication regroupe différentes formations, de la licence au master, ainsi que divers Diplômes d'Université (DU). Au sein de cet institut, j'ai effectué une bonne part de mes enseignements dans les formations ayant trait à la *Communication* et la *Création Infographique*. D'autre part, il m'a également été permis d'intervenir au Magistère d'Informatique et Modélisation à l'École Normale Supérieure de Lyon grâce à un échange de service (cours ASI, Analyse et Synthèse d'Images), et d'animer un stage de formation continue à l'Université Lyon 2 (Internet, HTML, Javascript).

1.4 Activités de recherche

1.4.1 Environnement et orientation générale

Le contexte général de mes activités de recherche est la *géométrie discrète*. Cette thématique s'intègre, au moins d'un point de vue historique, dans l'analyse de formes dans des images numériques. En effet, de nombreux 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). La seconde se base sur une transposition de ces théorèmes et mesures dans l'espace discret. Ces différentes re-définitions donnent lieu au paradigme mathématique et informatique qu'est la *géométrie discrète*.

Au fil du temps et des développements, de nombreuses connexions ont été mises en évidence avec de nombreuses thématiques des mathématiques (géométrie, topologie, arithmétique, théorie des nombres, combinatoire, analyse non-standard,...), du traitement du signal, et de l'informatique (algorithmique, géométrie algorithmique, complexité,...).

Si le spectre de la géométrie discrète est très large, les travaux que j'ai menés portent sur l'analyse de solutions algorithmiques à des problèmes de géométrie discrets. Plus précisément, nous nous intéressons à des algorithmes *exacts* dans le sens où nous cherchons à contrôler l'exactitude des calculs par l'utilisation d'arithmétique exacte ; mais aussi l'exactitude de l'analyse de complexité : nous cherchons des algorithmes efficaces dont le coût algorithmique est borné de la manière la plus fine possible.

Pour arriver à ces deux objectifs, nous avons utilisé des outils de divers domaines :

- d'une arithmétique dédiée à la manipulation numérique de quantités continues, faisant le lien entre *supercouverture* et *arithmétique d'intervalles* (chapitres 4 et 6) ;

- de l’algorithmique et de l’arithmétique entière ou plus précisément des outils de programmation linéaire entière pour l’analyse de certains objets discrets (*préimage*, convexes discrets,...), voir chapitre 5 ;
- de la géométrie algorithmique dans les développements liés aux transformations en distance ou à l’extraction de l’axe médian (chapitres 7 et 8) ;
- de l’algorithmique de manière transversale à tous les chapitres et plus précisément concernant les preuves de NP-complétude de la segmentation en plans discrets (section 3.4.2 et chapitre 6) ou de la minimalité de l’axe médian (chapitre 8).

1.4.2 Collaborations internationales

- R. KLETTE, PR, Centre for Image Technology and Robotics, University of Auckland, Nouvelle-Zelande : collaboration dans le cadre de l’analyse d’estimateurs de longueur discrets (voir [?] et [?]) et sur l’analyse des plans discrets [BCK07] ;
- V. BRIMKOV, PR, SUNY Buffalo State College, Etats-Unis : collaboration dans le cadre de l’analyse de plans discrets [BCK07][CB06] ;
- S. SVENSSON, Centre for Image Analysis, Uppsala Universitet, Suède : Analyse de courbes discrètes 3D dans une application d’analyse de fibres de papier [?].

1.4.3 Collaborations nationales extérieures au LIRIS

- A. MONTANVERT, PR, UPMF-IUT2 et laboratoire GIPSA-lab, UMR 5216 : collaboration dans le cadre des transformations en distance, de l’extraction d’axe médian [CM07], et dans le projet de coordination de l’ouvrage Hermès [CMC07] ;
- J.-M. CHASSERY, DR CNRS, laboratoire GIPSA-lab, UMR 5216 : collaboration liée à l’analyse de *plus grand convexe inclus dans une forme 2D* et aux notions de dualité/polarité [CSD⁺05][?, ?] ;
- M. COUPRIE, PR, Laboratoire A2SI, ESIEE : collaboration dans le cadre de la définition d’une fonction bissectrice basée sur le diagramme de Voronoï discret d’une forme [CCZ07] ;
- E. ANDRES, PR, Laboratoire SIC, Poitiers : collaboration dans le cadre des travaux de MARTINE DEXET sur la polyédrisation réversible d’objets discrets [DCA06] ;
- J.-B. BRZOSKA, ING. HDR, Centre d’Étude de la neige, Météo-France, Grenoble : Projet *Neige* dont l’objectif était l’analyse géométrique de micro-échantillons de neige ;
- J.-P. REVEILLÈS, PR, laboratoire LLAIC, Clermont-Ferrand : collaboration pour l’analyse des cercles discrets [CGRT04][?]
- I. DEBLED-RENESSON, MCF, laboratoire LORIA, Nancy : collaboration pour l’analyse d’estimateurs de longueur 3D [?].

1.4.4 Animation de la recherche

1.4.4.1 IAPR Technical Committee “Discrete Geometry”

Depuis octobre 2006, je suis responsable du *Technical Committee 18¹* de l'*International Association on Pattern Recognition²* (IAPR). Ce comité international a pour objectif de faciliter les interactions et les collaborations entre chercheurs travaillant en géométrie discrète ou dans des domaines connexes (le TC regroupe actuellement 69 chercheurs). Pour cela, un site web regroupant des ressources (documents, codes, jeux de données, ...) a été créé. De plus, le TC participe à l’organisation de la conférence majeure en géométrie discrète : *Discrete Geometry for Computer Imagery* (DGCI). Pour ce travail, je suis aidé de deux

1. <http://www.cb.uu.se/~tc18/>

2. <http://www.iapr.org>

vice-responsables : STINA SVENSSON (Dr. Centre for Image Analysis, Uppsala, Sweden) en charge des relations avec DGCI et de RENETA BARNEVA (Pr., Kasling Lecturer, Department of Computer Science State University of New York at Fredonia, USA), en charge de la diffusion et de l'*internationalisation* des activités en géométrie discrète.

Sur la période 2003-2006, je participais déjà au TC18 en tant que co-responsable avec ANNICK MONTANVERT (Pr, Université Pierre Mendès-France, Grenoble).

1.4.4.2 14th IAPR International Conference on Discrete Geometry for Computer Imagery

Lors du DGCI de 2006 à Szeged en Hongrie, à la suite de notre proposition nous avons été choisi par le *Steering Committee* de la conférence pour organiser la 14^e édition en avril 2008 au laboratoire LIRIS à Lyon³. Le comité d'organisation se compose d'ISBELLE SIVIGNON et de moi-même en tant que co-responsables principaux, ainsi que de LAURE TOUGNE, FLORENT DUPONT, JEAN-MARIE BECKER et de SERGE MIGUET.

1.4.4.3 Organisation de manifestations nationales

- **JIG 2006** “Journées Informatique et Géométrie”⁴ : 60 participants, organisées au LIRIS en collaboration avec VALÉRIE BERTHÉ (DR-CNRS, LIRMM) et LAURENT VUILLEN (Pr., Univ. de Savoie). Cette manifestation portait sur les thèmes de la géométrie algorithmique, de la géométrie discrète, de la synthèse et de l'analyse d'images.
- **Groupe de Travail Géométrie Discrète** : organisation de plusieurs réunions à Lyon du groupe de travail géométrie discrète regroupant une trentaine de membres.

1.4.4.4 Participation à des comités de programme et de lecture

Participation à des comités de programme :

- Discrete Geometry for Computer Imagery (DGCI) 2005
- International Workshop on Combinatorial Image Analysis (IWCIA) 2004
- International Workshop on Combinatorial Image Analysis (IWCIA) 2006

Participation à des comités de lecture :

- relecteur pour les revues :
 - IEEE Trans. on Pattern Analysis and Machine Intelligence
 - IEEE Trans. on Image Processing
 - Discrete Applied Mathematics
 - Pattern Recognition
 - Pattern Recognition Letters
 - Computer and Graphics
 - Image Vision & Computing
 - Pattern Analysis & Applications
- relecteur régulier des conférences DGCI, IWCIA et International Conference on Pattern Recognition (ICPR) ;

1.4.4.5 Participation à des jurys de thèse

- JULIEN RICARD : “Indexation et recherche d’objets 3D à partir de requêtes 2D et 3D”, LIRIS, Lyon, suite à la co-direction de la thèse (voir section 1.5.1), décembre 2005.
- MARTINE DEXET : “Architecture d’un modeleur géométrique à base topologique d’objets discrets et méthodes de reconstruction en dimensions 2 et 3”, SIC Poitiers, examinateur, directeur : ERIC ANDRÈS, décembre 2006.

3. <http://liris.cnrs.fr/dgci2008>

4. <http://liris.cnrs.fr/jig2006>

- FRANÇOIS DE VIELLEVILLE : “Analyse de parties linéaires des objets discrets pour l’extraction de caractéristiques géométriques”, LaBRI, Bordeaux, examinateur, directeurs : ACHILLE BRAQUELAIRE et JACQUES-OLIVIER LACHAUD, juillet 2007.

1.5 Encadrement

1.5.1 Encadrement d’étudiants en thèse

ANNÉE	NOM	TITRE DE LA THÈSE	ENCADREMENT.
05–08	Antoine Vacavant	Généralisation de la géométrie discrète aux grilles isothétiques irrégulières	L. Tougne & D. Coeurjolly
02–05	Julien Ricard	Indexation d’objets 3D à partir de requêtes 2D et 3D	A. Baskurt & D. Coeurjolly

- ANTOINE VACAVANT : cette thèse est financée par une allocation de recherche dans le cadre de l’ACI “Jeunes Chercheurs”. Le sujet porte sur la généralisation des outils classiques que l’on trouve en analyse et traitement d’images (filtrage, convolution, analyse de volumes et de contours *via* des outils de géométrie discrète, ...) sur des structures isothétiques irrégulières. ANTOINE VACAVANT a débuté son doctorat en octobre 2005 et est co-encadré par LAURE TOUGNE (Pr, Université Lumière Lyon2, laboratoire LIRIS) avec une date de soutenance prévue en 2008 (voir section 1.6).
- JULIEN RICARD : co-encadrement avec ATILLA BASKURT (Pr, INSA de Lyon, laboratoire LIRIS), date de soutenance : 8 décembre 2005. Dans le cadre du projet exploratoire RNRT “SEMANTIC 3D”⁵ comprenant des partenaires universitaires (LIRIS-Lyon, ENIC-Lille, INT-Paris, Eurecom-Sophia Antipolis) et industriels (Renault, TGS), les objectifs de ces travaux étaient de fournir des outils mathématiques et algorithmiques pour l’indexation de modèles 3D et la recherche dans une base de données à partir de requêtes 2D et 3D. Cette thèse a donné lieu à 2 publications dans des revues internationales et 6 conférences internationales.

1.5.2 Encadrements de stagiaires

Master Recherche 2ième année

- MARS 2006 – JUILLET 2006 : co-encadrement du stage de recherche de LAURENT JOSPIN avec FLORENT DUPONT (50 %). Le stage portait sur l’extraction de primitives géométriques de haut niveau (troncs de cônes, tétraèdres de boules,...) à partir de l’axe médian dans un objectif de compression et de transmission progressive d’objets discrets.
- MARS 2005 – JUILLET 2005 : encadrement du stage de recherche de LOUTFI ZERRARGA. Le stage portait sur l’élaboration d’un algorithme de reconstruction réversible de courbes isothétiques irrégulières. Cette problématique s’inscrit directement dans le projet “Jeunes Chercheurs” soutenu par le Ministère de la Recherche. L’objectif était de construire une approximation polygonale d’une courbe discrète irrégulière en maintenant une contrainte qui nous permettait, à tout moment, de retrouver la courbe discrète originale à partir de la reconstruction polygonale.
- MARS 2004 – JUILLET 2004 : co-encadrement du stage de recherche de CÉDRIC CHEVARIER avec FLORENT DUPONT (50%). Le stage portait sur l’utilisation de l’axe médian pour la transmission progressive d’objets discrets.

Master 1ère année

5. <http://www.semantic-3d.net>

- 2003–2004 : stage de JEAN-CHRISTOPHE BERNARD portant sur l’analyse d’algorithmes de calcul exact d’enveloppe convexe de points répartis sur la surface d’une sphère et application à l’estimation de courbure gaussienne.

Encadrements de stagiaires niveau Licence

- 2004–2005 : encadrement d’étudiants en TER (Travaux d’Etude et de Recherche) dans le cadre de l’ACI Jeunes Chercheurs :
 - ROMAIN EHRET et THIBAULT VALON : étude des librairies d’arithmétique d’intervalles et d’arithmétique affine.
 - CHRISTOPHE BEGON et THIERRY DUBOEUF : création d’une librairie de construction et manipulation de structures QuadTree.

1.6 Activités contractuelles

Dans cette section, je reprends les différents projets et contrats auxquels j’ai participé.

- **ANR Blanc GeoDib** ”Géométrie discrète pour les objets bruités” (Sept. 2006 - Sept. 2010) (qualité : **membre**, responsable : ISABELLE DEBLED-RENESSON, LORIA) : l’objectif de ce projet est de développer une géométrie des objets discrets bruités en utilisant de nouvelles techniques issues de la géométrie discrète et en se détachant des méthodes classiques comme la logique floue ou encore les techniques de filtrage utilisées en traitement d’images. Partenaires : LABRI, LIRIS, LLAIC, LORIA.
- **ACI Jeunes Chercheurs GeomDiGIT** ”Généralisation de la Géométrie Discrète aux Grilles Irrégulières IsoThétiques” (Sept. 2004 - Sept. 2007) (qualité : **responsable scientifique**) : dans certaines applications, les données acquises ou à traiter ne sont pas exactement structurées sur des grilles régulières. Par exemple, les grilles sous-jacentes aux appareils d’acquisition en imagerie médicale 3D sont généralement anisotropiques. Plus précisément, il y a un facteur d’échelle entre les axes de la grille, on parle alors d’une grille avec élongation. Il est donc important d’étendre à ce type d’espace discret tous les outils d’analyse géométrique disponibles dans le modèle discret usuel. Plutôt que de proposer des solutions ad-hoc dépendant du type de grille du capteur, nous nous proposons dans le cadre de cette ACI d’avoir une approche globale pour la redéfinition de ces outils sur ces structures. Partenaires : LIRIS, TSI UMR 5516 (Université Jean Monnet, Saint-Etienne). C’est dans le cadre de ce projet que je co-encadre la thèse d’ANTOINE VACAVANT avec LAURE TOUGNE (voir section [1.5.1](#)).
- **Projet RNRT Semantic 3D** “Service d’Echange et de MANipulation (Tatouage, Indexation et Compression)” (Nov. 2002 - Nov. 2005) (qualité : **membre**, responsable : ATILLA BASKURT) : le projet SEMANTIC-3D avait pour objet l’exploration de techniques et d’outils préalables à la réalisation de nouveaux services pour l’exploitation de contenu 3D à travers le Web. De nouvelles techniques de compression, d’indexation et de tatouage de données 3D ont été développées et mises en œuvre dans une application industrielle prototype. Partenaires : LIRIS, ENIC Lille, EURECOM, INT, Renault, TGS. C’est dans le cadre de ce projet que j’ai participé à la co-direction de la thèse de JULIEN RICARD (voir section [1.5.1](#)).

1.7 Activités administratives et collectives

1.7.1 Au sein de l’Université Claude Bernard Lyon 1

Je suis membre élu du Conseil Scientifique de l’Université depuis l’automne 2006. A ce titre et en dehors de ma participation aux conseils, je suis membre du groupe de travail

“Diffusion de la culture scientifique, technique et industrielle”. Je suis aussi membre du conseil du Service Commun de la Documentation (SCD).

1.7.2 Au sein du laboratoire

Je participe aux commissions suivantes du laboratoire :

- *commission Communication* : plus précisément, je suis membre de l'équipe webmaster du laboratoire et participe donc à la mise en place de l'infrastructure logicielle liée au site web du laboratoire (gestion des membres, gestion des publications,...) ;
- *commission Relations Internationales* : dont l'objectif était de constituer un état des lieux des relations internationales du laboratoire afin de proposer des mesures et incitations pour le développement d'une politique internationale. Un rapport préliminaire a été transmis et conclut donc la première étape de l'activité de cette commission.

1.7.3 Au sein d'autres établissements

Je suis membre suppléant de la commission de spécialistes 26/27/61 de l'Université Lumière Lyon2.

1.8 Publications

Synthèse des travaux :

- Un ouvrage édité et un manuscrit de thèse déposé
- 10 publications dans des revues internationales avec comité de lecture
- 23 articles dans des actes de conférences internationales avec comité de lecture
- 6 articles dans des actes de conférences nationales avec comité de lecture

1.8.1 Ouvrages

1.8.2 Chapitre dans ouvrage

Dans le cadre du traité IC2 *Géométrie discrète et images numérique*, j'ai rédigé les chapitres suivants :

- “Eléments de base”, D. COEURJOLLY, A. MONTANVERT et J.-M. CHASSERY ;
- “Distances discrètes”, E. THIEL et D. COEURJOLLY ;
- “Représentation par axe médian”, D. ATTALI, D. COEURJOLLY et E. REMY ;
- “Descripteurs de formes et moments géométriques”, D. COEURJOLLY et A. MONTANVERT.

1.8.3 Revues internationales avec comité de lecture

1.8.4 Conférences internationales

1.8.5 Conférences nationales

Chapitre 1. Rapport d'activités

Chapitre 2

Introduction générale

Pour définir la géométrie discrète en une phrase, on pourrait sûrement dire que cela consiste à résoudre des problèmes géométriques sur des données définies sur des grilles régulières. Si avec cette définition nous englobons nos activités sans peine, il nous faut quand même préciser le contexte.

Si nous reprenons un peu l'histoire et les grands noms de ce domaine, nous pouvons identifier deux grandes façons d'aborder la géométrie discrète : l'approche « pragmatique » et l'approche « constructiviste ». Si ce dernier terme a une signification très particulière en mathématique, nous l'interprétons dans ce qui suit que dans une version plus épistémologique.

L'approche pragmatique peut être résumée par le fait qu'il existe des données structurées sur grilles discrètes et qu'il faut bien des outils efficaces pour traiter des problèmes de géométrie sur ces données. Dans ce cadre, nous retrouvons tous les développements ayant démarrés dans les années 70 avec l'essor des écrans matriciels et des premiers besoins de synthèse, d'analyse et de traitement d'images numériques.

Concernant la partie synthèse, il a fallu définir des outils de conversion d'un tracé vectoriel (point, segment,...) vers un tracé discret (ensemble de points d'une grille que nous allumons ou non). La problématique est donc de convertir des objets continus vers des objets discrets, on parle alors de processus de discréttisation. Pour l'analyse d'image, la structuration régulière, induite soit par les capteurs, soit issue d'un processus de ré-échantillonage, est intimement liée aux images numériques. La définition des outils permettant de faire des mesures, de traiter ou d'analyser des objets contenus dans des images, rentre dans le contexte de la géométrie discrète. Par la suite, les capteurs tridimensionnels ou intégrant une composante temporelle sont apparus et des besoins d'outils de dimension supérieure se sont fait sentir.

Il serait cependant faux de croire que l'approche pragmatique n'est à mettre en relation qu'avec des problématiques d'imagerie moderne. Dans son *Traité pour les astronomes*, les développements de JEAN BERNOULLI [Ber71] s'intègrent parfaitement dans ce contexte : pour résoudre un problème de calcul d'interpolation linéaire de positions d'étoiles pour la construction d'éphémérides, il s'est rendu compte qu'en considérant des approximations entières pour les positions, l'interpolation mettait en œuvre des structures périodiques et donc simplifiait grandement les calculs. Beaucoup plus tard, ces observations ont été prouvées [Chr75] et on peut mettre en relation ce fait avec les notions de périodicité dans les droites discrètes. Là aussi on peut qualifier cette approche de pragmatique : les nombres entiers et l'arithmétique se sont imposés d'eux-mêmes pour résoudre efficacement un problème donné.

Plus proche de nous, cette technique d'arithmétisation des processus a été grandement utilisée pour résoudre efficacement des problèmes dans de nombreux domaines comme l'extraction de coupes dans des images volumiques médicales [CMC07, chap. 17], ou encore pour coder efficacement des transformations d'images [CMC07, chap. 7]. Ces éléments sont des hauts faits d'arme de la géométrie discrète.

Les objets et propriétés géométriques qui ont été définies au cours du temps dans ce cadre forment un paradigme géométrique complet. L'objectif est généralement de se rapprocher, autant que possible, de la géométrie euclidienne classique.

En parallèle de cette vision de la géométrie discrète, nous pouvons aussi considérer une approche constructiviste qui, de manière très schématique, consiste à dire que le discret n'est pas un sous-échantillonage élégant du réel mais que le réel est construit à partir du discret. Il est cette fois impossible de dresser un réel historique de cette approche démarquant dans la Grèce antique : ce mode de construction du réel par combinaisons d'éléments discrets correspond à une thématique entière des mathématiques. Les liens entre cette construction presque axiomatique du réel et la géométrie discrète ont été présentés par l'école d'analyse non standard de Strasbourg (HARTONG, REEB, REVEILLÈS,...). L'objectif est à l'origine d'offrir une modélisation des nombres réels non plus basée sur une représentation illimitée (ou comme la limite d'une suite) mais sur des entiers (standard et non-standard). Ensuite HARTONG par exemple [Har89], montre que des analyses infinitésimales peuvent être menées en n'utilisant que les propriétés arithmétiques des entiers, les axiomes de PEANO et le prédicat *St()* de l'analyse non-standard.

Ces premiers liens entre géométrie discrète et cette approche constructiviste des réels peuvent être attribués à REVEILLÈS [Rev91]. Dans un souci d'implémentation d'algorithmes géométriques sur des ordinateurs, l'analyse non standard offre, à une échelle donnée, une modélisation géométrique exacte sur laquelle les erreurs sont contrôlables et qui reste valide aux échelles supérieures (notion de Théorie Idéale Discrète de REVEILLÈS). Pour illustrer ce propos, les travaux de REVEILLÈS portaient initialement sur la résolution rapide d'équations différentielles en exploitant à la fois les outils théoriques de l'analyse non standard mais aussi les spécificités algorithmiques induites par les ordinateurs (représentation finie,...). Dans ce contexte, la droite discrète est apparue intrinsèquement dans l'analyse sans lien *a priori* avec le continu (résolution numérique de $y' = c$). Dans un second temps bien sûr, des liens ont été présentés entre cet objet donné par une analyse constructive et la discréttisation d'une droite réelle classique.

Une analyse épistémologique plus approfondie serait très intéressante à mener et permettrait sûrement de mieux appréhender la place de la géométrie discrète dans ou à côté de la géométrie classique.

Que ce soit par la théorie constructiviste du modèle discret ou par une approche pragmatique, la géométrie discrète a maintenant une maturité qui fait d'elle un véritable espace de recherche théorique ou appliquée avec ses objets, ses théorèmes et ses spécificités. Elle ne peut donc être réduite exclusivement ni à une sous-classe de la géométrie euclidienne, ni à un cas particulier de la géométrie algorithmique, ou encore une série d'outils bas-niveau pour le traitement et l'analyse d'images. La géométrie discrète offre des solutions à certains de ces domaines, et c'est là une grande force, mais possède une existence propre.

Quelle est maintenant la place de nos travaux et de ce manuscrit dans ce contexte ? Dans leur finalité, nos développements trouvent écho dans les deux domaines : analyse théorique par exemple d'objets fondamentaux (modèles de grille, droites, plans cercles,...), mais aussi outils d'analyse volumique pour la caractérisation de formes (transformation en distance, axe médian, ...). Le fil conducteur est l'algorithmique : que ce soit sur des aspects théoriques internes à la géométrie discrète ou pour proposer des outils, l'objectif ultime est l'algorithme. Ces derniers se doivent d'être efficaces et sans erreur.

Pour arriver à cet objectif, nous exploitons de nombreux résultats d'autres domaines : de la géométrie algorithmique car les finalités sont proches, mais aussi de l'arithmétique ou de la théorie des nombres pour tenir compte des spécificités discrètes des objets.

Organisation du manuscrit

Dans un premier temps, nous allons revenir dans le chapitre 3 sur quelques définitions élémentaires et sur tout un ensemble de problématiques liées à la géométrie discrète mais

qui ne seront pas forcément reprises par la suite. Dans le chapitre 4, nous présentons les différents modèles analytiques que l'on peut définir en géométrie discrète et nous présentons une approche basée sur l'arithmétique d'intervalles pour caractériser l'un d'entre eux.

Le chapitre 5 détaille les différentes solutions algorithmiques pour la reconnaissance des objets élémentaires de la géométrie discrète que sont les droites, les plans et hyperplans, et les cercles discrets. Ce chapitre est l'illustration parfaite de l'usage de résultats d'un grand nombre de domaines (géométrie algorithmique, programmation linéaire, arithmétique,...) pour pouvoir proposer des solutions algorithmiques efficaces.

Le chapitre 6 met en œuvre des objets fondamentaux dans une problématique de reconstruction réversible de contours en dimension 2 et 3. Plus précisément, nous cherchons à construire un représentant euclidien d'un ensemble de points discrets (par exemple une courbe polygonale pour un ensemble de pixels), réversible dans le sens où la discréétisation de cet objet sera exactement l'ensemble de points discrets.

Dans le chapitre 7 nous abordons d'autres problèmes classiques de la géométrie discrète que sont la transformation en distance ou l'extraction d'axe médian. Là encore, nous verrons que les liens très forts entre géométrie discrète et géométrie algorithmique nous offre une nouvelle façon d'aborder les problèmes.

Le chapitre 8 s'intéressera à l'optimisation et aux applications liées à la représentation d'objets discrets par leur axe médian. On s'intéressera notamment à la question de la minimalité de l'axe médian.

Enfin, nous présentons une conclusion générale ainsi que quelques perspectives en fin de manuscrit. L'annexe A liste les différentes réalisations logicielles que j'ai pu développer au cours de mes analyses.

Chapitre 3

Notions de base

Sommaire

3.1	Image, image numérique et support d'image	18
3.2	Espace discret et connexité	18
3.2.1	Pavages et maillages	18
3.2.2	Voisinages	21
3.3	Objets, courbes, surfaces	23
3.3.1	Définitions classiques	23
3.3.2	Définitions pour les grilles isothétiques irrégulières	24
3.4	Eléments géométriques	25
3.4.1	Objets discrets : droites, plans, cercles, etc.	25
3.4.2	Reconstruction et polyédrisation réversible d'objets discrets . . .	26
3.4.3	Distances discrètes, transformée en distance et axe médian . . .	27
3.5	Éléments de complexité	28

3.1 Image, image numérique et support d'image

Une image représente une scène ou un objet, soit après acquisition *via* un capteur, soit par extraction d'une vue après modélisation par des primitives. Nous considérons dans cet ouvrage les diverses variétés d'images qui peuvent en résulter. Nous tenons également compte des représentations en plusieurs dimensions, et en particulier des images en trois dimensions, de plus en plus courantes dans de nombreuses applications. Cependant, pour la plupart des notions présentées, nous les explicitons tout d'abord en deux dimensions.

Que la scène observée soit réelle ou simulée, nous nous intéressons, en dimension deux, à sa projection sur le plan associé à un capteur (plan de focalisation par exemple, dans le cas d'une caméra ou d'un appareil photographique). Dans ce plan, l'image est de nature analogique et correspond à une distribution d'intensités lumineuses, notée $f : \mathbb{R}^2 \rightarrow I$. Pour faciliter la manipulation, le stockage et l'analyse de l'image par ordinateur, on utilise couramment une représentation de f sur une partition dénombrable de \mathbb{R}^2 , notée \mathcal{P} . On parle alors d'*image numérique* ou encore d'*image discrète*. Plus précisément, si un point P est associé à chacune des cellules V_P de la partition \mathcal{P} d'aire S_P , certains capteurs construisent une représentation discrète de f , notée F , définie par :

$$F(P) = \frac{1}{S_P} \int \int_{V_P} f(x, y) dx dy$$

En d'autres termes, l'intensité de la cellule V_P est associée au point P et contient la somme de toutes les contributions de f à la région du plan définie par la cellule. Comme nous le verrons dans la suite, ce processus est appelé *processus de discréétisation*. Généralement, l'espace d'intensité I associé à f (couleur, énergie, etc.) est aussi transformé pour pouvoir être représenté sur un ordinateur, en un espace I' . On parle alors de fonction de transfert et une modélisation plus réaliste du processus de création d'une image numérique peut être donnée par :

$$F(P) = g \left(\frac{1}{S_P} \int \int_{V_P} f(x, y) dx dy \right)$$

avec :

$$\begin{aligned} f &: \mathbb{R}^2 \rightarrow I \\ F &: \mathcal{P} \rightarrow I' \\ g &: I \rightarrow I' \end{aligned}$$

La décomposition du plan de projection en cellules dénombrables définit le *support de l'image numérique*. Les mosaïques, assemblages de tesselles teintées une à une, peuvent être considérées comme les prémisses des représentations discrètes (voir figure 3.1). Cependant, la disposition et la taille des tesselles épouse les motifs plutôt que de respecter les propriétés géométriques d'un support figé.

Usuellement, les images numériques sont fournies sur un support régulier, avec un espace I' correspondant à trois canaux (Rouge, Vert, Bleu, noté RVB) où chaque canal contient une intensité sur un intervalle $[0, 255]$ (voir figure 3.2).

3.2 Espace discret et connexité

3.2.1 Pavages et maillages

Par la suite, nous appelons *espace discret* un pavage du plan en dimension deux, ou plus généralement de l'espace en dimension trois ou plus. Nous appelons aussi *point discret* le centre de gravité de chaque cellule dans le pavage considéré. Dans ce qui suit, nous représentons graphiquement l'espace discret, soit par le pavage, soit par le maillage issu de

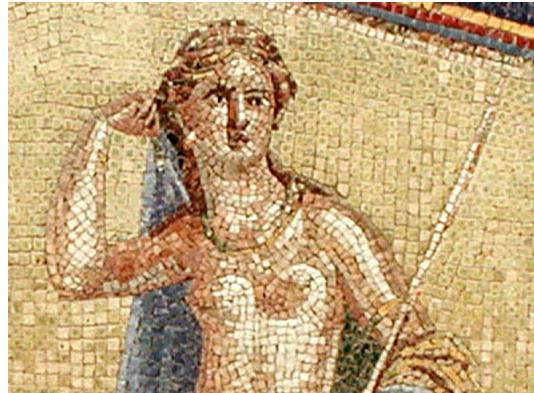


FIGURE 3.1 – Détail d'une mosaïque à Herculaneum, représentant Amphitrite.

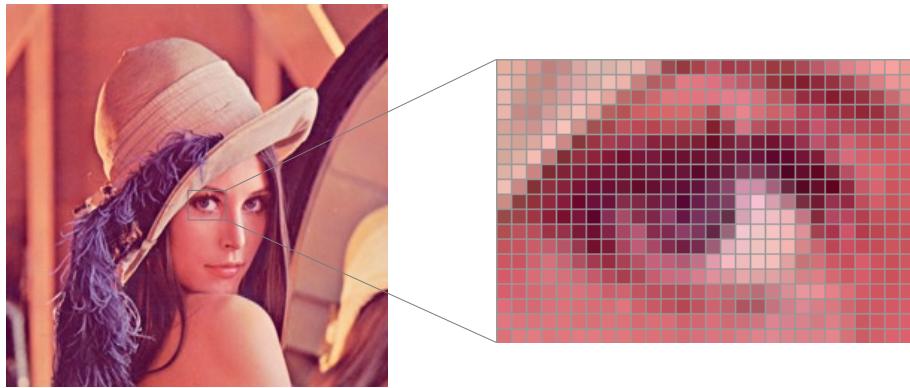


FIGURE 3.2 – Représentation d'une image sur un support discret régulier.

ce pavage. Le maillage correspond à un graphe dont les sommets sont les points discrets et dont les arêtes représentent les adjacences entre cellules, éléments du pavage. La figure 3.3 présente les trois pavages et maillages possibles en dimension deux.

Certaines partitions de l'espace peuvent être étudiées, donnant lieu à des pavages qui ne sont pas forcément réguliers [GS87]. Par ailleurs d'autres représentations de l'espace sont parfois utilisées. Par exemple en dimension trois, les représentations BCC et FCC (*body-centered cubic, face-centered cubic*), issues de l'organisation des cristaux, possèdent des propriétés particulières de symétrie et de densité, qui peuvent être exploitées pour le codage et le traitement d'images.

De par la physique des capteurs, le pavage par des carrés est le plus usuel, même si, comme nous le verrons par la suite, ses caractéristiques topologiques ne sont pas toujours les plus simples. En dimension supérieure à deux, les seuls espaces discrets réguliers sont ceux engendrés par des cubes multi dimensionnels. Les points discrets associés à ces espaces sont donc des points de \mathbb{Z}^d pour la dimension d .

En terme de codage, l'espace discret engendré par le pavage avec des carrés offre une représentation matricielle directe : une image correspondra à une partie de \mathbb{Z}^2 . Dans le cas des pavages par des triangles ou par des hexagones, un codage particulier est nécessaire, mais nous pouvons toujours considérer la grille \mathbb{Z}^2 (voir figure 3.4). L'intérêt d'une représentation matricielle est qu'elle permet un adressage direct des éléments (un couple de coordonnées pour chaque point discret) ainsi qu'une extraction rapide des points discrets voisins.

Dans le chapitre 4, nous nous intéressons à une classe de pavages et de maillages un peu différente : les *grilles isothétiques irrégulières*. Cette classe consiste à considérer un pavage

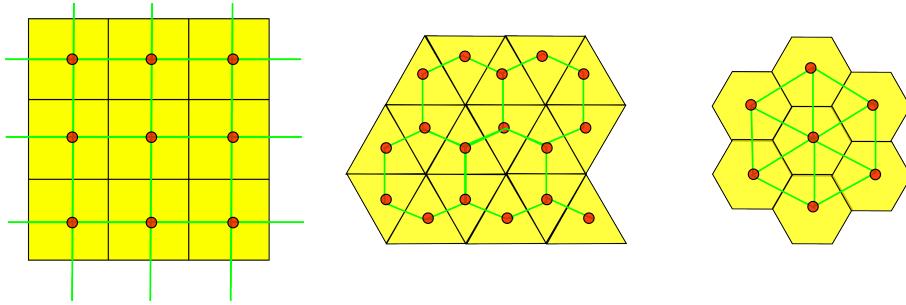


FIGURE 3.3 – Pavages (et maillages) réguliers en dimension deux par cellules carrées, triangulaires et hexagonales.

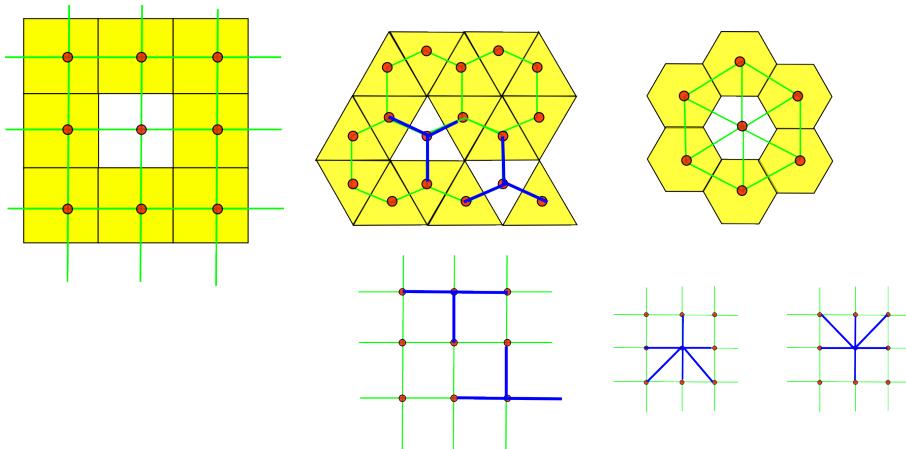


FIGURE 3.4 – Représentation et codage des pavages réguliers 2D sous forme matricielle.

avec des carrés et à relâcher les contraintes sur la taille des cellules, ainsi que sur la position de leur centre. Supposons une cellule définie comme un rectangle donné par un centre $p \in \mathbb{R}^2$, une largeur l_x et une hauteur l_y ($l_x, l_y \in \mathbb{R}$). En fonction des contraintes que nous imposons sur p , l_x et l_y , nous pouvons définir des *pavages isothétiques* comme un ensemble de cellules tel que l’intersection de tout couple de cellules est soit vide, soit de dimension inférieure ou égale à 1. Notons que ces pavages peuvent ne pas pavier le plan mais dans nos applications, ils pavieront généralement un domaine rectangulaire (voir chapitre 4).

Par exemple (voir figure 3.5), si $p \in \mathbb{Z}^2$ et $l_x = l_y = 1$, nous définissons le pavage régulier par carré (\mathbb{D}). En considérant maintenant des rectangles de taille uniforme $l_x = \lambda$ et $l_y = \mu$ ($\lambda, \mu \in \mathbb{R}$) avec pour centres $p = (\lambda i, \mu j)$ ($(i, j) \in \mathbb{Z}^2$), nous pouvons caractériser les grilles avec élongation, très utilisées en imagerie médicale car de nombreux appareils d’acquisition n’ont pas la même résolution suivant toutes les dimensions.

De manière similaire, si nous considérons non plus des contraintes sur la forme des cellules isothétiques indépendamment les unes des autres, mais des contraintes globales comme des règles de subdivision d’un rectangle initial (par des droites horizontales ou verticales), nous pouvons caractériser des pavages issus de grilles hiérarchiques comme le *quadtree* (\mathbb{Q}) [Sam90]. Si nous considérons des regroupements/fusions de cellules issues d’un pavage régulier, nous pouvons définir les grilles adaptatives (\mathbb{A}) (voir par exemple [Hav01] pour une étude de ces différentes grilles dans le cas de l’accélération d’une lancer de rayons en synthèse d’images).

L’intérêt d’avoir une écriture unifiée, sous le nom de grilles isothétiques irrégulières, de ces pavages est de pourvoir dériver des définitions d’objets ou des algorithmes génériques

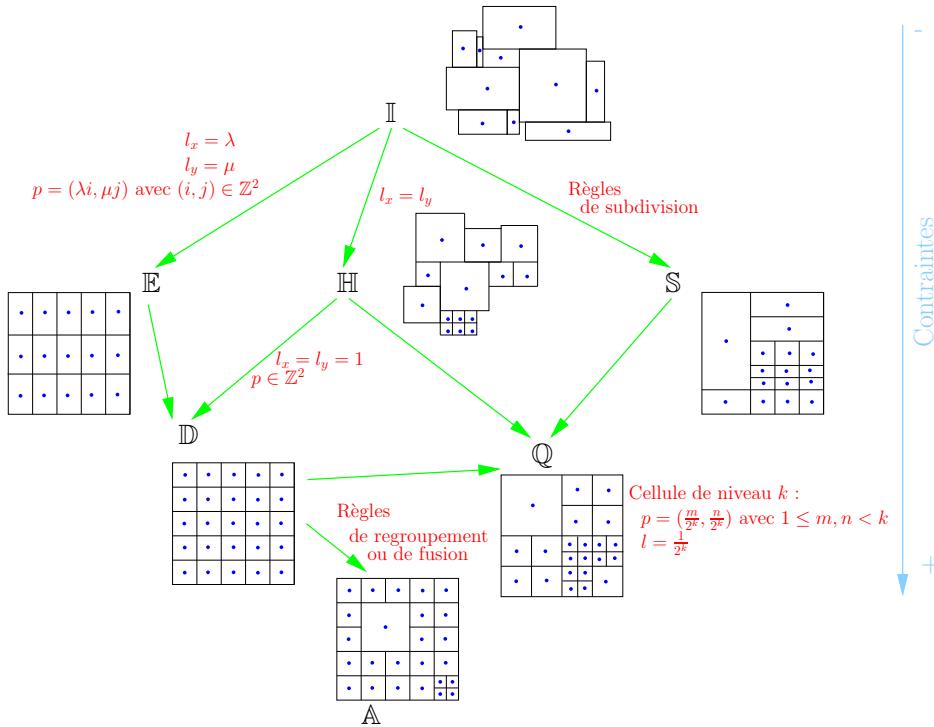


FIGURE 3.5 – Structuration possible de quelques grilles isothétiques irrégulières.

transversaux à tous ces modèles. Le chapitre 4 présente quelques-uns de ces outils.

3.2.2 Voisinages

Sur un espace discret, nous introduisons des notions de *voisinage* permettant de construire des graphes d'adjacence ou, dit plus simplement, de détecter si deux pixels sont voisins ou non. Si nous considérons l'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 sommets. Plus formellement, deux points A et B de \mathbb{Z}^2 , de coordonnées respectives (x_A, x_B) et (y_A, y_B) sont 4-*voisins* (ou 4-*adjacents*) si :

$$|x_A - x_B| + |y_A - y_B| = 1.$$

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

$$\max(|x_A - x_B|, |y_A - y_B|) = 1.$$

En dimension trois, nous pouvons introduire les notions de 6-, 18- ou 26-*voisinage* en considérant les adjacences par faces, arêtes et sommets (voir figure 3.6). Ces adjacences sont données formellement dans le tableau 3.1.

adjacence	caractérisation
6	$ x_A - x_B + y_A - y_B + z_A - z_B = 1$
18	A et B sont 26-voisins et $ x_A - x_B + y_A - y_B + z_A - z_B \leq 2$
26	$\max(x_A - x_B , y_A - y_B , z_A - z_B) = 1$

TABLE 3.1 – Caractérisation des adjacences en dimension trois.

FIGURE 3.6 – En 3D, différents voisinages du cube central avec le pavage régulier.

Dans ce qui suit, on parle d' α -adjacence, pour $\alpha \in \{4, 8, 6, 18, 26\}$.

Une écriture unifiée de ces différentes relations d'adjacence s'exprime sous la forme de (r) -voisinage, généralisable aux dimensions supérieures : deux points de \mathbb{Z}^d sont (r) -voisins si chaque coordonnée diffère au plus de 1 et qu'au moins r coordonnées sont égales. Alors, les connexités 4 et 8 en dimension 2 sont notées respectivement (1)- et (0)-voisinage. De même, les connexités 6, 18 et 26 en 3D s'écrivent respectivement (2), (1) et (0) (voir table 3.2 pour une comparaison des différentes notations). Une manière plus simple de considérer le (r) -voisinage est de regarder la dimension de l'intersection de deux pixels : deux pixels (carrés unités fermés) A et B sont (1)-voisins si l'intersection de A et B contient un élément de dimension 1 (un segment en 2D). De même, A et B sont pixels sont (0)-voisins si leur intersection contient des éléments de dimension supérieure à 0 (un point ou un segment en 2D).

(r) -voisinage	α -voisinage
0	$3^d - 1$
...	...
$d - r$	$\sum_{i=d-r}^{d-1} \frac{d!}{i!(d-i)!} 2^{d-i}$
...	...
$d - 2$	$2d^2$
$d - 1$	$2d$

TABLE 3.2 – Lien entre les notations de voisinage pour un pavage par hypercubes.

Cette illustration géométrique de la (r) -adjacence est généralisable aux dimensions supérieures de la grille discrète, mais aussi au cas des grilles isothétiques irrégulières (voir figure 3.7).

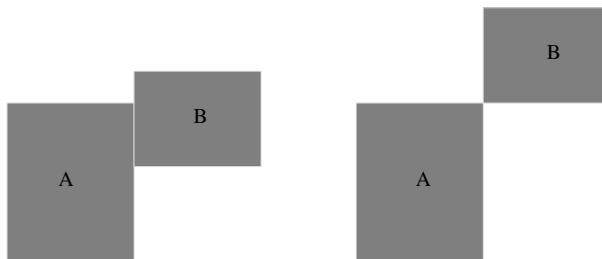


FIGURE 3.7 – A gauche, A et B sont (1)- et (0)-adjacents, à droite, A et B sont (0)-adjacents mais pas (1)-adjacents.

3.3 Objets, courbes, surfaces

3.3.1 Définitions classiques

Dans l'espace discret, nous décrivons les objets élémentaires utilisés par la suite. Nous notons $r \in \{0, 1, 2\}$ la relation d'adjacence considérée sur la grille discrète \mathbb{Z}^d pour $d \in \{2, 3\}$. Ces relations d'adjacence nous permettent tout d'abord de définir un (r) -chemin sur l'espace discret :

DÉFINITION 3.1 ((r)-chemin) Soit un ensemble X de points discrets et une relation de (r) -adjacence. Un (r) -chemin (dans X) joignant deux points A et B de X est une séquence $\pi = (A_0, \dots, A_n)$ de pixels de X telle que $A_0 = A$, $A_n = B$ et A_i est (r) -voisin de A_{i-1} pour tout $i = 1, \dots, n$.

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

DÉFINITION 3.2 ((r)-objet) Soit un ensemble X de points discrets et une relation de (r) -adjacence. X est un (r) -objet si pour tout couple A et B de X , il existe un (r) -chemin dans X .

En d'autres termes, un (r) -objet est une composante connexe de points discrets au sens de la (r) -adjacence. La notion de (r) -chemin décrite ci-dessus est très générale. Nous restreignons cette notion pour pouvoir introduire des propriétés topologiques plus complexes.

DÉFINITION 3.3 ((r)-courbe fermée) Soit π un (r) -chemin. π est une (r) -courbe fermée si tous les éléments de π ont exactement deux points (r) -voisins dans π .

Finalement, si nous déconnectons une (r) -courbe fermée, nous définissons tout simplement une (r) -courbe.

DÉFINITION 3.4 ((r)-courbe) Soit π un (r) -chemin. π est une (r) -courbe si, pour tous les éléments $\{A_i\}$ de π , A_i a exactement deux points (r) -voisins, sauf A_0 et A_n qui n'en ont qu'un (A_0 et A_n sont alors les extrémités de la courbe).

L'introduction des courbes lève des problèmes topologiques. En effet, une propriété *souhaitable* sur une (r) -courbe fermée dans \mathbb{Z}^2 serait que l'on puisse définir un intérieur et un extérieur à cette courbe. Ces définitions sont liées à la notion de propriété de JORDAN en géométrie différentielle classique.

Pour illustrer cette notion, prenons l'exemple d'un processus de remplissage très simple : étant donné un pixel *source* (pixel noir dans la figure 3.8), le remplissage d'une région se fait en coloriant tous les voisins selon une connexité r de (0) ou (1). Sur la courbe donnée dans la figure 3.8, nous voyons que, parmi les deux connexités possibles du remplissage, seul le remplissage en (1)-connexité remplit effectivement la courbe discrète sans déborder. En utilisant la (0)-connexité, tout l'espace discret est rempli, ce qui peut être critique dans certaines applications. La propriété de JORDAN permet de formaliser cette notion de *perméabilité* d'un contour. L'énoncé de cette propriété ainsi que les techniques permettant la caractérisation des objets discrets la vérifiant est un problème très classique en géométrie discrète. D'une manière générale, deux grandes approches sont considérées dans ces formalisations : la première consiste à définir les contours d'objets discrets comme des séquences de points discrets. La seconde se base sur une décomposition cellulaire de la grille (voir figure 3.9) et définit le contour d'un objet comme les éléments de dimension inférieure (à la dimension de l'espace) séparant les pixels ou les voxels de l'objet, de ceux du complémentaire [CMC07, chap. 3].

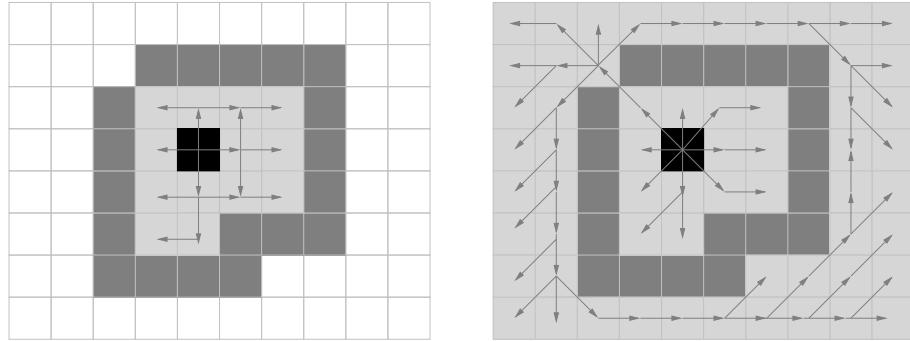


FIGURE 3.8 – Illustration d’un remplissage d’objet à partir du pixel noir en (1)–connexité (*gauche*) et en (0)–connexité (*droite*). Les flèches illustrent un exemple de propagation selon la connexité considérée.

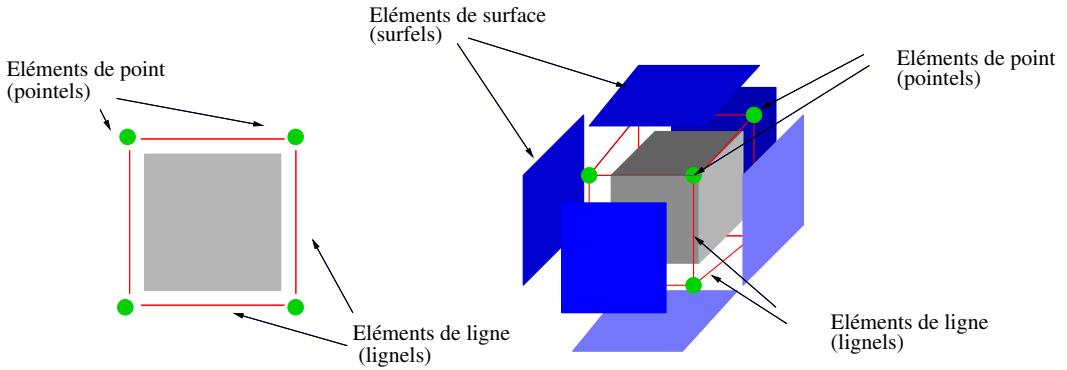


FIGURE 3.9 – Décomposition d’un pixel et d’un voxel en cellules de dimensions inférieures.

3.3.2 Définitions pour les grilles isothétiques irrégulières

Dans le cas des grilles isothétiques irrégulières, les définitions précédentes sont tout à fait valides pour la (0)– et la (1)–adjacences définies sur ces cellules. Nous avons donc de manière triviale, des (r)–objets ou encore des (r)–courbes fermées dans ces espaces (voir figure 3.10).

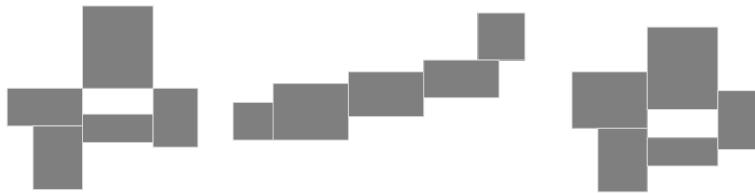


FIGURE 3.10 – Exemples d’objets topologiques sur une grille isothétique irrégulière : (*de gauche à droite*) un (1)–objet (mais non (0)–objet), une (0)–courbe et une (0)–courbe fermée.

Pour retrouver une définition de contour sur ces espaces, nous considérons un pavage de cellules étiquetées avec un label “objet” ou “fond”. Ensuite, nous pouvons nous baser sur la notion de surface de JORDAN sur un graphe bicolore orienté [Her98] : les sommets du graphe sont les cellules coloriées “objet” ou “fond”, et les arêtes sont données explicitement par le graphe d’adjacence des cellules pour une certaines connexités.

3.4 Eléments géométriques

Sur les pavages réguliers, nous avons pour l'instant caractérisé les objets discrets selon leurs propriétés topologiques, ou selon le processus de discréétisation d'un objet réel dont ils sont issus.

Nous pouvons cependant définir un paradigme géométrique de manière intrinsèque à la grille discrète. Ce paradigme regroupe des définitions d'objets géométriques élémentaires (points, droites, plans, segments, cercles, etc.), des propriétés entre ces objets, ainsi que des algorithmes permettant de les manipuler efficacement.

3.4.1 Objets discrets : droites, plans, cercles, etc.

Dans le paradigme de la géométrie discrète, il est nécessaire de définir des objets géométriques de base. Généralement, deux approches complémentaires sont possibles ; nous les illustrons avec la définition d'une droite discrète :

- définition par discréétisation : on définit la droite discrète comme étant le résultat d'une discréétisation, selon un certain processus, d'une droite réelle ;
- définition intrinsèque : on définit la droite discrète comme étant l'objet géométrique qui valide la version discrète de certaines propriétés géométriques que possède la droite réelle.

Souvent ces deux processus se rejoignent et décrivent le même objet. Reprenons la définition par discréétisation, s'il est évident que la figure 3.11.c ne peut pas être un morceau de droite discrète, le fait que la courbe 3.11.b n'en soit pas un non plus illustre le fait que ces objets possèdent une structure intrinsèque. Plus précisément, la droite discrète de pente rationnelle, tout comme le plan discret dans une certaine mesure, possède des structures de régularité et de périodicité faisant le lien entre ces objets et des résultats fondamentaux en arithmétique, théorie des nombres, théorie des mots, etc. Une bibliographie très dense existe sur ces éléments, [KR04b, BCK07] présentent un nombre conséquent de références.

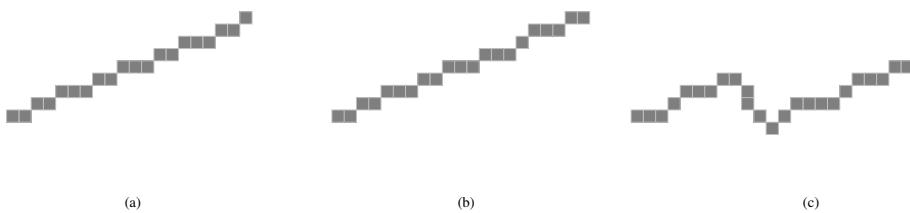


FIGURE 3.11 – Seule la courbe (a) est un segment discret selon les définitions usuelles des droites discrètes.

En utilisant la droite discrète dans notre modèle géométrique, nous pouvons d'ores et déjà nous rendre compte que les axiomes géométriques d'EUCLIDE seront à revoir. En effet, l'intersection, en termes de points et donc de pixels communs, entre deux droites discrètes non parallèles peut être composée d'un point, d'aucun point, d'un ensemble connexe de points ou encore d'un ensemble non connexe de points. De même, deux droites discrètes parallèles peuvent partager une infinité de points sans être confondues. Il faut donc repenser les propriétés d'intersection et de parallélisme [Vee99, Vee02, SDC04].

Ce problème, posé par les axiomes d'EUCLIDE, illustre parfaitement la problématique de la géométrie discrète : le lien avec l'objet ou le concept continu sous-jacent est à considérer avec beaucoup de précautions. Parfois nous aurons des constructions ou des résultats intrinsèques au modèle discret sans aucun lien avec le continu, parfois nous définirons nos objets discrets comme étant des approximations les plus précises possibles d'objets continus.

Dans le chapitre 4 nous revenons sur les modèles de discréétisation, et sur les liens discrets/continus.

Étant donnée une définition d'un objet fondamental comme la droite discrète, une question récurrente est d'obtenir un algorithme de reconnaissance : considérons un ensemble de pixels (pour l'exemple de la droite discrète), un algorithme de reconnaissance doit décider s'il existe une droite discrète contenant l'ensemble des pixels considéré. De plus, comme la définition de la droite discrète est paramétrée par des quantités (pente, ordonnée à l'origine), nous attendons de l'algorithme qu'il nous retourne aussi un paramétrage valide pour l'ensemble (dans le cas où ce dernier est bien un sous-ensemble de droite discrète). Bien sûr, ces techniques vont exploiter les propriétés internes des objets à reconnaître afin d'obtenir des algorithmes rapides [KR04b, BCK07]. Le chapitre 5 présente une analyse théorique et algorithmique pour certains de ces objets.

Par définition des objets fondamentaux comme la droite discrète, un paramétrage valide conduit généralement à la définition d'une droite réelle associée avec la propriété que la discréttisation de celle-ci corresponde exactement aux pixels de la droite discrète. Notons qu'il existe une infinité de droites réelles se discréttisant dans une même droite discrète.

3.4.2 Reconstruction et polyédrisation réversible d'objets discrets

Une problématique classique en géométrie discrète consiste à convertir une représentation en extension (un ensemble de pixels d'un contour discret) en une représentation en compréhension (courbe polygonale par exemple). Bien évidemment, ce changement de représentation se doit d'être réversible dans le sens où, si l'on discréttise l'objet continu (courbe polygonale), nous devons obtenir l'objet discret initial.

Dans ce contexte, nous voyons bien que les objets fondamentaux vont être des éléments de base pour résoudre ce problème. Nous utiliserons en effet les algorithmes de reconnaissance de droite discrète pour décomposer un contour en segments. De même, nous pourrons utiliser les algorithmes de reconnaissance de plans discrets pour la facettisation de surface d'objets discrets 3D (voir figure 3.12).

Si les problèmes paraissent simple, leurs résolutions ne sont pas triviales : même si la définition des objets fondamentaux permet d'assurer la réversibilité le long du segment dans le cas 2D par exemple, le positionnement des sommets nécessite une analyse fine. En dimension 3, le placement des sommets et des arêtes est tout aussi difficile.

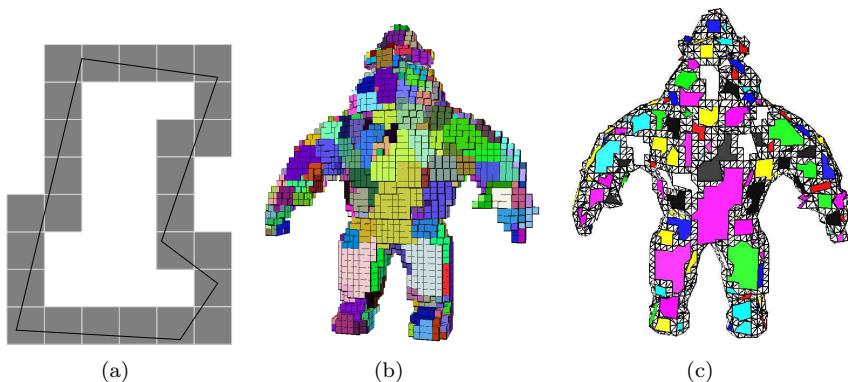


FIGURE 3.12 – Exemple de reconstruction réversible en dimension 2 et 3.

Généralement, trois approches sont envisagées (illustrées dans le cas 3D) :

- approche *Top-Down* : dans un premier temps, nous allons décomposer la surface en morceaux de plan maximaux selon un ensemble d'heuristiques (voir figure 3.12-(b)). Ces morceaux peuvent donc définir une infinité de facettes réelles et dans un second temps, nous cherchons à calculer les facettes permettant un *recollement* de la surface [SDC03, SC03, Siv04, SDC05] ;

- approche *contrainte* : la construction de la surface et la reconnaissance de plan discret se font ici en parallèle. Pour cela, la reconnaissance est *contrainte* par la surface polygonale en cours de construction afin d'obtenir la réversibilité sur les sommets et les arêtes, ainsi que d'avoir la correction topologique [Dex06, DCA06]. Un inconvénient est que la propagation des *contraintes* a tendance à *accumuler* les erreurs et donc à produire des polyèdres très irrégulières, avec des petites facettes ;
- approche *Bottom-Up* : l'idée est toujours de commencer par une décomposition en morceaux de plan, mais nous partons d'une facettisation primaire élémentaire, comme celle donnée par un algorithme de *Marching-Cubes* [LC87, LM00, NY06], que nous simplifions (fusion de facettes) en intégrant l'information de la segmentation en plans discrets [CDJS06]. Un élément clé de ce type de méthodes et de maintenir, pour chaque opération locale de simplification de la surface, les propriétés de réversibilité et de correction topologique (figure 3.12-(c)). La section 6.3 reviendra plus en détail sur cette technique.

Notons que la qualité de la segmentation en morceaux de plan discret, en terme de nombre de morceaux, influence grandement la suite de la reconstruction. Cette influence est directe pour les approches *Top-Down* et *Bottom-Up*, mais elle se retrouve aussi dans les approches *contraintes*. Sur ce problème, nous avons cependant démontré un résultat théorique sur le fait que la segmentation en un nombre minimal de morceaux est NP-difficile dans le cas général [SC06] (voir section 3.5 pour les notions liées à la NP-complétude).

Dans les illustrations de la figure 3.12, les objets discrets sont *simples* : nous avons une courbe (au sens de la section 3.3.1) en dimension 2 et une surface simple (une seule composante connexe, sans trous, etc.) en dimension 3. Dans le cas de structures discrètes plus complexes avec des jonctions par exemple, les problèmes sont plus compliqués.

Dans le chapitre 4, nous revenons sur ces problèmes de reconstruction en dimension 2 sur des formes complexes, et en dimension 3.

Cette dernière thématique est à rapprocher du domaine du traitement d'images, et plus particulièrement du traitement de documents, qu'est la *vectorisation* (voir par exemple [HT06] pour plus de références). Notons que dans ce cadre applicatif, la réversibilité est une contrainte rarement prise en compte. L'effort se porte sur des définitions statistiquement robustes aux bruits plutôt qu'à des représentations exactes.

3.4.3 Distances discrètes, transformée en distance et axe médian

Un autre domaine détaillé dans le chapitre 7 porte sur l'analyse volumique d'objets discrets par l'utilisation de la transformation en distance. Cette transformation consiste à étiqueter tous les points d'un objet discret avec leur distance minimale au complémentaire ; elle est d'une grande utilité pour représenter les objets discrets par des unions de boules, définissant ainsi l'*axe médian* d'une forme discrète (voir figure 3.13).

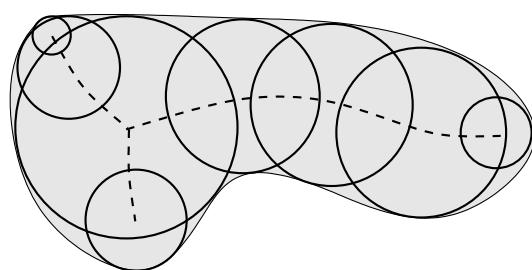


FIGURE 3.13 – Illustration de la représentation d'une forme par union de disques. Les centres des disques (pointillés) représentent schématiquement l'axe médian de la forme.

Pour cela, nous devons spécifier la distance que l'on considère. Etant donnés deux points

discrets P et Q , deux grandes catégories d'approches existent. La première consiste à manipuler la distance euclidienne directement en considérant le vecteur \overrightarrow{PQ} ou encore le carré de la distance, $d_e^2(P, Q)$. La seconde consiste à approximer la distance euclidienne par des masques locaux, donnant lieu aux *distances de chanfrein* [Bor86, Thi01]. Ainsi, en considérant une famille de déplacements élémentaires pondérés, nous pouvons construire un plus court chemin entre P et Q et estimer la distance entre ces points (voir figure 3.14).

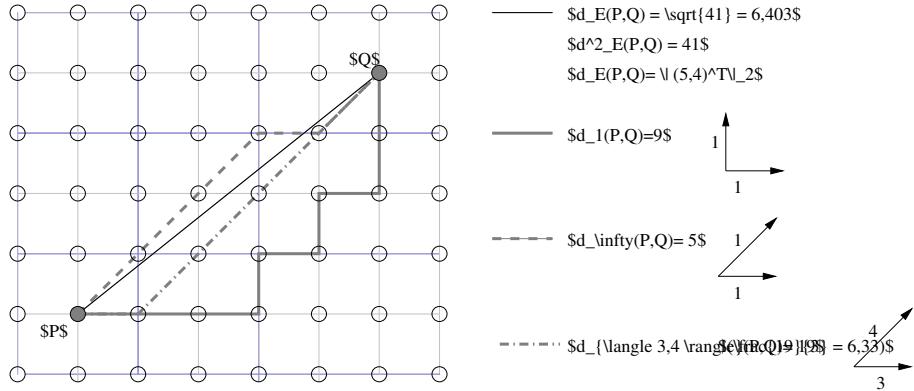


FIGURE 3.14 – Illustration de distances discrètes entre deux points discrets. Les chemins correspondent à des plus courts chemins pour la distance considérée (les déplacements élémentaires et leurs poids sont indiqués par les flèches).

Généralement, les approches fondées sur une approximation en distance de chanfrein étaient préférées, de par la simplicité des algorithmes sous-jacents (transformation en distance, extraction de l'axe médian,...). Néanmoins, nous montrons dans le chapitre 7 que des algorithmes très efficaces existent pour la métrique euclidienne.

3.5 Éléments de complexité

Dans cette section, nous présentons rapidement les notions de complexité que nous utiliserons par la suite. Plus précisément, nous reprenons le schéma classique de preuve de NP-complétude que nous utilisons dans le chapitre 7. Dans ce qui suit, nous considérons un modèle générique de *machine déterministe à accès aléatoire*, à processeur unique [CLR90]. Une description rapide de ce modèle affecte un coût unitaire pour toutes les opérations arithmétiques, ainsi que pour la comparaison de nombres. Pour quantifier la complexité d'un algorithme, nous utilisons généralement une analyse au pire cas, paramétrée par la taille des données n en entrée de l'algorithme. Nous considérons aussi les notations asymptotiques de complexité $O(g(n))$ définie de la manière suivante : si $f(n) = O(g(n))$, il existe une constante c et un nombre n_0 tel que pour tout $n > n_0$, $f(n) < cg(n)$. Nous disons aussi que $g(n)$ est une *borne supérieure asymptotique* pour $f(n)$. De manière similaire, nous définissons $\Omega(g(n))$ comme une *borne inférieure asymptotique* pour $f(n)$. Si $g(n)$ est une borne asymptotique inférieure et supérieure, nous notons $\Theta(g(n))$ la *borne approchée asymptotique* pour $f(n)$ [CLR90].

Ainsi, les notations précédentes nous permettent de quantifier la complexité d'un algorithme. Par extension, nous pouvons quantifier la complexité d'un problème. Si un problème a une complexité de $O(g(n))$, il existe nécessairement un algorithme résolvant le problème ayant une complexité $O(g(n))$. La quantification des problèmes, indépendamment des algorithmes les implémentant, devient formellement intéressante lorsque nous obtenons des preuves de borne inférieure ou asymptotique approchée $\Theta(g(n))$ pour un problème : dans ce cas, nous garantissons qu'il n'existe aucun autre algorithme ayant une complexité asymptotique moindre répondant au problème initial.

Pour mieux caractériser les algorithmes, nous cherchons à classifier ces derniers en classes de complexité. Plus précisément, nous allons nous intéresser à la classe P des algorithmes dont la résolution se fait en temps polynomial (complexité en $O(g(n))$ où $g(n)$ est un polynôme dont le degré ne dépend pas de n), mais aussi à une classe d'algorithmes pour laquelle un doute subsiste sur sa nature (NP-complétude). Dans ce qui suit, nous considérons des problèmes de décision, c'est-à-dire des problèmes prenant une entrée de taille n et retournant uniquement VRAI ou FAUX. Par abus de langage, un problème est vérifiable (ou résoluble) s'il existe un algorithme vérifiable (ou résoluble) implémentant ce problème.

Plus formellement :

DÉFINITION 3.5 (P) *Un problème de décision est dans la classe P s'il est résoluble sur une machine déterministe en temps polynomial.*

Le temps d'exécution de l'algorithme dépend du *codage* avec lequel l'entrée est donnée à l'algorithme. Nous ne détaillons pas ces points mais si nous changeons le codage d'une instance du problème de manière polynomiale (la *transformation* est calculable en temps polynomial), cela ne change pas la classe de complexité du problème [CLR90].

DÉFINITION 3.6 (NP) *Un problème de décision est dans la classe NP¹ si la validation d'un solution au problème est calculable sur une machine déterministe en temps polynomial. On dit alors que le problème est vérifiable en temps polynomial.*

De manière ensembliste, nous pouvons déduire de ces définitions que $P \subset NP$. Le problème classique porte sur l'existence de problèmes dans NP, qui ne seraient pas dans P. En d'autres termes, est-ce que $P = NP$? Si nous ne savons pas répondre à cette question, nous pouvons *lier* tous ces problèmes entre eux par la notion de NP-complétude. Pour cela, il faut tout d'abord présenter la *réduction polynomiale* : tout comme le recodage des entrées présenté plus haut, un problème \mathcal{P} peut être *réduit* en temps polynomial dans un autre problème \mathcal{P}' s'il existe une transformation polynomiale de **toutes** les instances de \mathcal{P} vers **un sous-ensemble** des instances de \mathcal{P}' .

DÉFINITION 3.7 (NP-complet) *Un problème \mathcal{P} est NP-complet si :*

- $\mathcal{P} \in NP$
- tous les problèmes de NP se réduisent polynomiallement en \mathcal{P} .

En d'autres termes, \mathcal{P} est au moins aussi difficile à résoudre que tous les autres problèmes de NP. Par conséquence, s'il existe un problème NP-complet pouvant être résolu en temps polynomial, alors tous les problèmes NP-complets pourraient être résolus en temps polynomial, et donc $P=NP$. L'existence d'un tel problème, ou la preuve de sa non-existence est un problème ouvert.

Pour prouver qu'un problème \mathcal{P} est NP-complet, le principe est plus simple que de considérer tous les problèmes NP, il faut :

- considérer un problème \mathcal{P}' prouvé comme étant NP-complet ;
- trouver une réduction polynomiale de toutes les instances de \mathcal{P}' vers un sous-ensemble d'instances de \mathcal{P} ;
- montrer que l'existence d'un algorithme pour résoudre \mathcal{P} sur ces instances reviendrait à résoudre \mathcal{P}' .

Ainsi, \mathcal{P} est au moins aussi difficile que \mathcal{P}' .

Le premier problème NP-complet est dû à COOK [Coo71] : considérons une formule booléenne satisfiable composée des opérateurs booléens ET (\wedge), OU (\vee), et NON (\neg) et de k variables (booléennes). Le problème de décision est le suivant : quelle que soit la

1. NP est l'acronyme de *Non-deterministic Polynomial* : il y a équivalence pour la définition de la classe entre “être **vérifiable** en temps polynomial sur une machine déterministe” et “être **résolvable** en temps polynomial sur une machine non-déterministe”.

formule booléenne satisfiable, existe-t-il une instanciation des variables permettant de valider la formule (problème SAT) ?

Dans le schéma de preuve de NP-complétude présentée ci-dessus, il est parfois plus commode de considérer les problèmes suivants :

- 3-SAT : les instances sont les formules ϕ qui sont des expressions logiques de m clauses sur n variables binaires en forme normale conjonctive d'ordre 3, par exemple :

$$\phi = (x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_4 \vee x_5) \wedge (\neg x_6 \vee x_3 \vee \neg x_5) \wedge \dots,$$

Le problème de décision est donc le suivant : existe-t-il une instanciation des variables permettant de satisfaire ϕ ?

- PLANAR-3-SAT : les instances de ce problème sont les formules ϕ qui sont des instances de 3-SAT particulières. Considérons le graphe dont les sommets sont les variables et les clauses. Les arêtes d'une variable vers une clause signifie que la variable apparaît dans la clause [JM95]. Cette instance est une instance de PLANAR-3-SAT si le graphe associé est planaire (voir figure 3.15). Le problème de décision ne change pas.

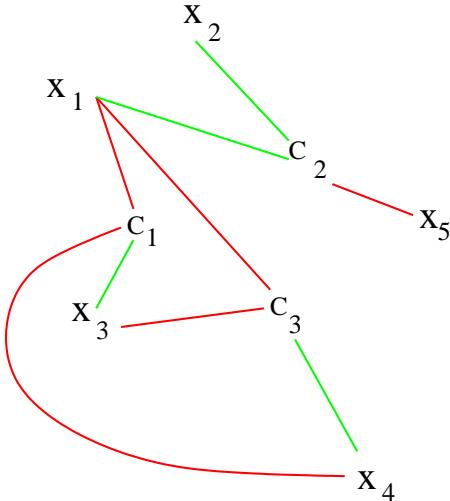


FIGURE 3.15 – Représentation planaire de l'instance de PLANAR-3-SAT : $\phi = (x_1 \vee \neg x_3 \vee x_4) \wedge (\neg x_1 \vee \neg x_2 \vee x_5) \wedge (x_1 \vee x_3 \vee \neg x_4)$. Les clauses c_1, c_2 et c_3 sont numérotées dans l'ordre de la formule et les deux types de liens correspondent à l'utilisation de la variable de manière négative ou positive.

- PLANAR-4-3-SAT : les instances de ce problème sont les formules ϕ , instances de PLANAR-3-SAT, telles qu'une variable n'apparaît qu'au plus 4 fois dans la formule (sous forme négative ou non). Le problème de décision est identique.

Généralement, ces problèmes NP-complets sont très utilisés pour la preuve de NP-complétude de problèmes géométriques. 3-SAT nous permet de construire des objets géométriques pour les *variables* et les *clauses* (schéma utilisé pour prouver la NP-complétude de la décomposition minimale en morceaux de plans discrets d'une surface discrète, voir [SC06]). PLANAR-3-SAT est très utilisé dans le cas d'algorithmes géométriques bi-dimensionnels (voir section 8.2). Enfin, PLANAR-3-SAT offre l'avantage de construire des schémas géométriques pour les variables plus simples car celle-ci ne sera *utilisée* que 4 fois au plus (sur la représentation en graphe, les sommets "variables" seront de degré au plus 4 et les sommets "clauses" de degré 3). notons que le plongement géométrique d'un graphe planaire peut se faire en temps linéaire et ce résultat est toujours valide lorsque le plongement se fait sur une grille [dPP90]

Finalement, si nous avons un problème d'optimisation, par exemple, "décomposer la

forme en un nombre minimal de morceaux ayant une certaine propriété”, la transformation en un problème de décision peut être donnée par la formule : “existe-t-il une décomposition de la forme avec k morceaux ayant une certaine propriété ?” avec k une variable libre dans l’énoncé. Usuellement, si le problème de décision associé à un problème d’optimisation est NP-complet, le problème d’optimisation sera dit *NP-difficile*.

Chapitre 4

Modèle analytique discret, arithmétique d'intervalles et supercouverture

Sommaire

4.1	Introduction	34
4.2	Modèles de discrétisation	34
4.2.1	Discrétisations basées sur le maillage	34
4.2.2	Discrétisations basées sur le pavage	36
4.2.3	Modèles analytiques discrets	36
4.3	Arithmétique d'intervalles et supercouverture	38
4.3.1	Éléments de base	39
4.3.2	Solveur en arithmétique d'intervalles	41
4.3.3	Lien avec un modèle analytique discret de supercouverture	42
4.4	Conclusion	45

4.1 Introduction

Dans la section 3.4.1, nous avons présenté les différentes approches pour la définition d'objets dans le modèle discret. Dans ce chapitre, nous détaillons quelques éléments concernant les processus de discréétisation.

Si ce chapitre contient des notions qui seront exploitées dans les chapitres suivants, nous présentons une vision alternative d'un de ces modèles, le modèle de supercouverture, en nous intéressant à une arithmétique particulière fondée sur les intervalles. De manière transversale, le problème est toujours le même : nous nous intéressons à la représentation et à la manipulation d'une description discrète et partielle d'un objet euclidien.

Le contexte est le suivant : nous considérons un objet continu F ou uniquement son bord, noté ∂F . Nous souhaitons construire l'ensemble des points discrets qui sont les *discréétisés*, soit de F (notez que F contient aussi ∂F), soit exclusivement de ∂F . Une analyse de ces différents modèles ainsi que d'autres solutions peuvent se trouver dans [JK97, OK95, And00, KR04a].

4.2 Modèles de discréétisation

Un processus de discréétisation se doit d'être invariant par translation entière, pour garantir un comportement identique de la discréétisation à tout endroit de l'espace discret. Une autre propriété que l'on retrouve dans tous les processus classiques concerne la stabilité : lorsque l'on perturbe localement l'objet euclidien, sa discréétisation ne doit être modifiée que localement.

D'autres propriétés sont discutées dans les paragraphes suivants, mais nous pouvons d'ores et déjà noter l'importance de la résolution de la grille pour que le processus de discréétisation préserve des propriétés vérifiées par la représentation continue (voir figure 4.1). Par exemple, il faudra choisir une résolution de grille dépendante de la *complexité* de l'objet continu pour que certaines propriétés topologiques se retrouvent dans le discréétisé. Plus précisément, on pourra lier la résolution, d'une part à la courbure maximale du contour de l'objet euclidien et d'autre part à son épaisseur locale [SK05, RT00b, TR02].

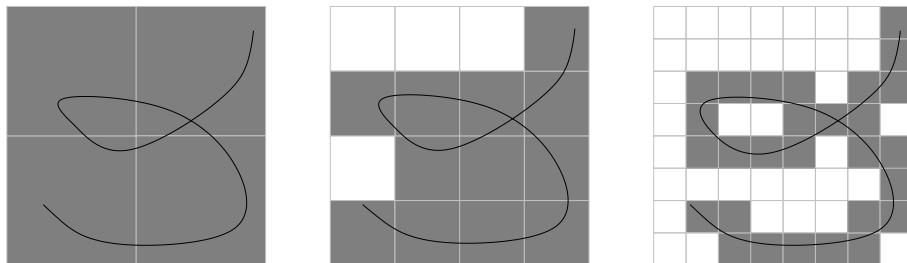


FIGURE 4.1 – Rôle de la résolution pour une cohérence entre la topologie de l'objet continu par rapport à l'objet discret, avec une discréétisation en supercouverture (voir paragraphe 4.2.3).

4.2.1 Discréétisations basées sur le maillage

Considérons dans un premier temps un objet continu F dont le bord possède la propriété de JORDAN (c'est-à-dire avec un intérieur et un extérieur). Le premier processus de discréétisation est dû à GAUSS pour une estimation d'aire de l'objet euclidien. Le principe est le suivant : la discréétisation de l'objet correspond à l'ensemble des points de coordon-

nées entières contenus dans F . Ce processus est aussi appelé *Object Boundary Quantization* (OBQ).

Dans le cas d'une discréétisation de courbe ne disposant pas de la notion d'intérieur, nous ne pouvons évidemment pas considérer la discréétisation de GAUSS ; d'autres schémas existent pour cela. Nous citons ici la discréétisation *Grid Intersect Quantization* (GIQ) qui approxime la courbe par l'ensemble des pixels *les plus proches* selon un certain critère : à chaque fois que la courbe croise une arête du maillage, deux pixels sont considérés et le plus proche de l'intersection selon la distance euclidienne fait partie de la discréétisation de la courbe (voir la figure 4.2). Le cas pathologique pour ce schéma est le cas où la courbe passe à égale distance des deux pixels. Il faut soit considérer les deux points (d'où l'apparition de *bulles* dans la discréétisation, voir section 4.2.2), soit faire un choix systématique et arbitraire (par exemple *le pixel d'abscisse ou d'ordonnée la plus petite*). Remarquons ici que ce choix rend le processus de discréétisation non invariant à une permutation des axes de la grille.

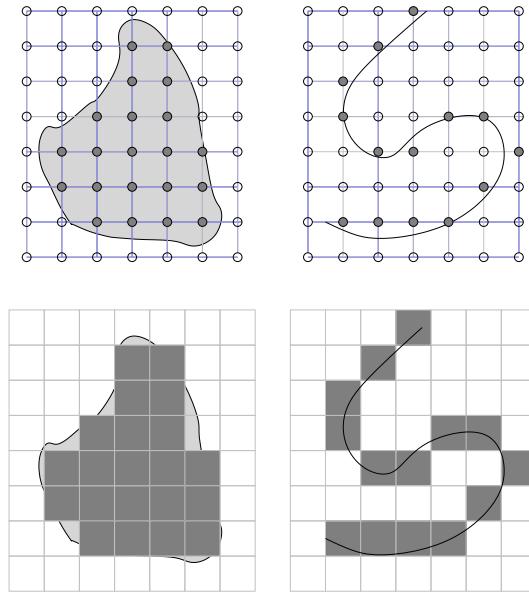


FIGURE 4.2 – Processus de discréétisation classiques sur le maillage (première ligne) ou sur le pavage (seconde ligne) : à gauche, discréétisation de GAUSS d'un objet et à droite, discréétisation GIQ d'une courbe.

Nous précisons maintenant des schémas de discréétisation spécifiques aux droites et plans discrets. Étant donnée une droite D dans le premier octant (pente positive inférieure à 1), la discréétisation GIQ de D peut être vue comme un processus d'arrondi à l'entier le plus proche. En effet, pour $D : y = \alpha x + \beta$ par exemple, les points de la discréétisation sont donnés par¹ :

$$\{(x_i, y_i) : y_i = [\alpha x_i + \beta], x_i \in \mathbb{Z}\},$$

Par abus de langage, la discréétisation :

$$\{(x_i, y_i) : y_i = \lfloor \alpha x_i + \beta \rfloor, x_i \in \mathbb{Z}\}$$

est appelée *discréétisation OBQ de D*. De même, la discréétisation :

$$\{(x_i, y_i) : y_i = \lceil \alpha x_i + \beta \rceil, x_i \in \mathbb{Z}\}$$

1. $[x]$ désigne l'arrondi de x à l'entier le plus proche x . De plus, $\lfloor x \rfloor$ (respectivement $\lceil x \rceil$) correspond à l'entier inférieur (respectivement supérieur) le plus proche de x .

est appelée *discrétisation BBQ (Background Boundary Quantization)* de D . La figure 4.3 illustre ces différents schémas.

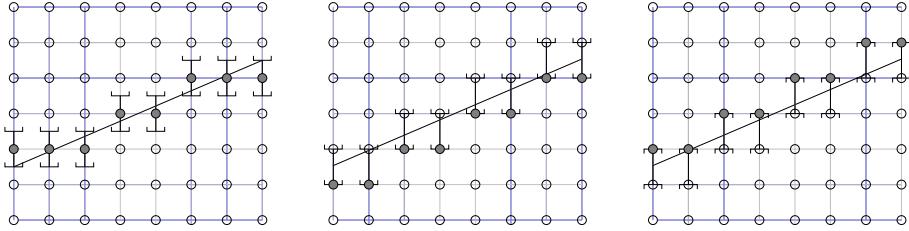


FIGURE 4.3 – Différents schémas de discrétisation spécifiques aux droites : (de gauche à droite) discrétisations GIQ, OBQ et BBQ. Les intervalles représentent les opérations d’arrondi.

4.2.2 Discrétisations basées sur le pavage

Nous présentons tout d’abord un schéma de discrétisation, appelé *supercouverture*, utilisant la représentation par pavage [OK95, CM91, And00] : la discrétisation d’un objet euclidien F en supercouverture, notée $\mathbb{S}(F)$ considère tous les pixels (fermés) intersectés par F (voir figure 4.4). Ce schéma, très simple, définit un processus de discrétisation valide quel que soit le type d’élément (courbe, objet, point, etc.). Il permet de plus de montrer des propriétés fondamentales utiles en modélisation, par exemple pour l’assemblage d’objets. Voici quelques-unes des propriétés de la discrétisation en supercouverture :

$$\begin{aligned}\mathbb{S}(F \cup G) &= \mathbb{S}(F) \cup \mathbb{S}(G), \\ \mathbb{S}(F) &= \bigcup_{p \in F} \mathbb{S}(p), \\ \mathbb{S}(F \cap G) &\subseteq \mathbb{S}(F) \cap \mathbb{S}(G), \\ \text{si } F \subseteq G \text{ alors } \mathbb{S}(F) &\subseteq \mathbb{S}(G).\end{aligned}$$

Ce processus est généralisable en dimension supérieure, il suffit de considérer les intersections de la forme avec les voxels ou les hypercubes. Cependant, cette discrétisation produit des objets parfois épais. En effet, si l’objet continu passe par un sommet du pavage (coordonnée demi-entière), toutes les cellules adjacentes à ce sommet font partie de la discrétisation ; on parle alors de *bulles* (voir figure 4.4). Ainsi, la discrétisation d’une droite ne sera pas une (1)–courbe comme on aurait pu le souhaiter mais un (1)–objet. Encore une fois, avec des conventions d’orientation dans le cas d’une discrétisation de structures linéaires, nous pouvons supprimer ces bulles. Le modèle issu de ces choix est appelé *modèle standard* [And02]. Dans ce schéma, la discrétisation d’une droite est bien une courbe (1)–connexe. Comme dans le cas du choix fait pour GIQ, l’orientation choisie dans ce modèle rend le processus non invariant aux symétries par rapport aux axes de la grille.

4.2.3 Modèles analytiques discrets

Dans la présentation précédente, nous pouvons généraliser les schémas de discrétisation en supercouverture et standard en définissant un *modèle analytique discret*. Pour cela, nous n’allons plus nous intéresser aux cellules du pavage qui sont intersectées par l’objet continu mais à l’intersection entre l’objet et des régions convexes incluses dans les pixels (voir figure 4.5).

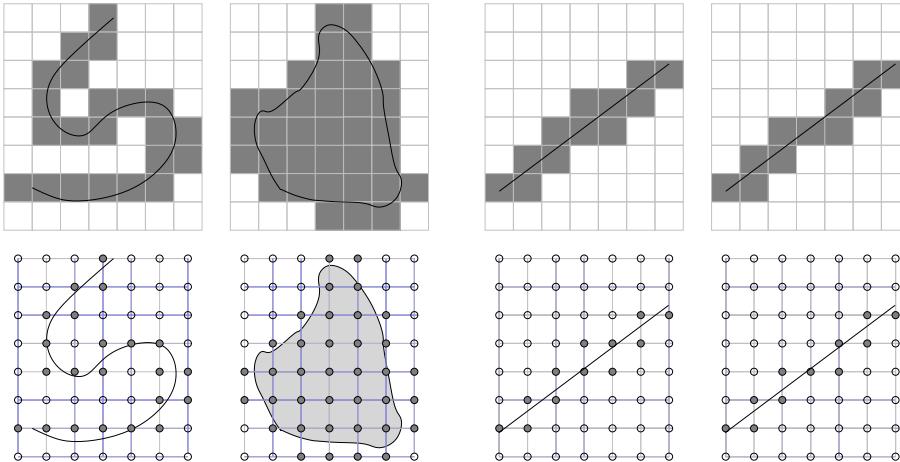


FIGURE 4.4 – Processus de discréétisation classiques sur le pavage (première ligne) ou sur le maillage (seconde ligne) : (de gauche à droite) discréétisation en supercouverture d'une courbe, discréétisation en supercouverture d'un objet, illustration d'une bulle pour la discréétisation d'un segment de droite, et discréétisation en modèle standard.

Si la région convexe est le pixel lui-même, nous retrouvons le modèle de supercouverture. Si la région est un losange centré, on parle de *modèle naïf fermé*. Enfin si la région est un disque, nous définissons le *modèle pythagoricien* [And00].

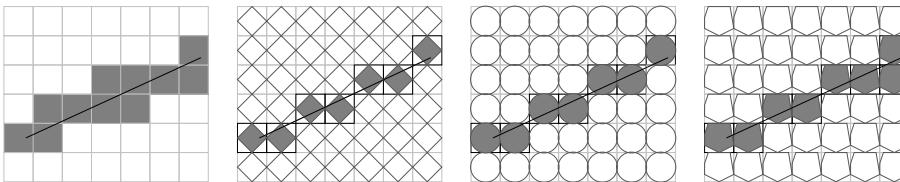


FIGURE 4.5 – Quelques exemples de modèles analytiques : supercouverture, naïf fermé, pythagoricien et engendré par un pentagone.

Formalisons ces modèles : nous considérons une région de base C et nous supposons que cette région correspond à une boule de rayon $\frac{1}{2}$ pour une certaine distance d . Par exemple, le carré correspond à la boule de rayon $\frac{1}{2}$ pour la distance $d_\infty(A, B) = \max\{|x_A - x_B|, |y_A - y_B|\}$, le losange correspond à la boule de rayon $\frac{1}{2}$ pour la distance $d_1(A, B) = |x_A - x_B| + |y_A - y_B|$. Nous notons M l'élément structurant engendré par la *réflexion* de C par rapport à son centre. Ces notions d'élément structurant et de réflexion viennent directement de la morphologie mathématique [Ser82]. Le modèle de discréétisation analytique \mathbb{A}_C d'un objet euclidien F est donné par trois définitions équivalentes :

$$\begin{aligned} \mathbb{A}_C(F) &= \{A \in \mathbb{Z}^d \mid C(A) \cap F \neq \emptyset\} \\ &= \{A \in \mathbb{Z}^d \mid d(A, F) \leq \frac{1}{2}\} \\ &= (F \oplus M) \cap \mathbb{Z}^d \end{aligned}$$

La première rejoint la définition que nous avons donnée précédemment, la seconde considère les points discrets à une certaine distance de la forme et la dernière nous donne une écriture relevant de la morphologie mathématique où \oplus correspond à la somme de MINKOWSKI [Ser82, Soi99] de l'élément structurant M et de la forme F . Pour cette dernière définition, nous considérons les points discrets contenus dans la forme continue $F \oplus M$ (voir figure 4.6).

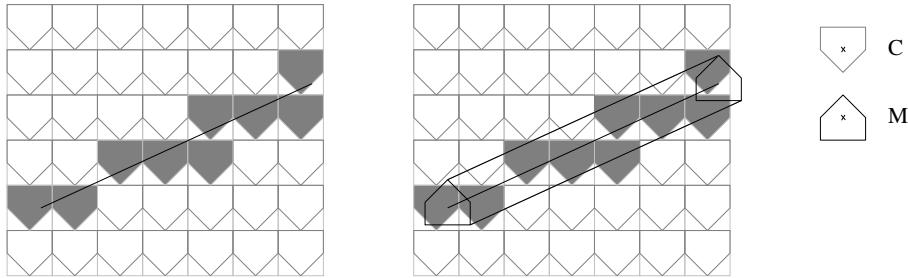


FIGURE 4.6 – Illustration des définitions d'un modèle analytique d'une forme quelconque : définitions par intersection avec les régions et par morphologie mathématique (C est la région et M sa réflexion par rapport au centre de C).

Sur ce modèle générique, nous retrouvons les propriétés :

$$\mathbb{A}_C(F \cup G) = \mathbb{A}_C(F) \cup \mathbb{A}_C(G), \quad (4.1)$$

$$\mathbb{A}_C(F) = \bigcup_{P \in F} \mathbb{A}_C(P), \quad (4.2)$$

$$\mathbb{A}_C(F \cap G) \subseteq \mathbb{A}_C(F) \cap \mathbb{A}_C(G), \quad (4.3)$$

$$\text{si } F \subseteq G \text{ alors } \mathbb{A}_C(F) \subseteq \mathbb{A}_C(G). \quad (4.4)$$

Enfin, étant donnée la topologie de l'objet F , des résultats peuvent être démontrés sur la topologie de sa discréttisation $\mathbb{A}_C(F)$ en considérant des propriétés vérifiées par la région C [And00, LW03].

4.3 Arithmétique d'intervalles et supercouverture

Dès les premiers pas de l'informatique théorique (voire des mathématiques discrètes) et même bien avant la réalisation *physique* des ordinateurs, s'est posé le problème de la représentation et de la manipulation de quantités réelles.

Généralement, plusieurs sous-problèmes peuvent être énoncés :

- Comment représenter toutes les quantités réelles ?
- Comment faire des calculs numériques sur ces représentations ?
- Quels contrôles peut-on mettre en place pour évaluer les approximations, s'il y a lieu, de ces calculs approchés par rapport aux résultats *exact*s ?
- Comment prendre une décision ou tester la *véracité* d'un prédicat basé sur le résultat d'un calcul ?

Plus simplement, les deux premiers points reposent sur la constitution d'une arithmétique. Le troisième point fait le lien entre les résultats approchés par l'arithmétique choisie et les résultats donnés par un modèle de calcul exact. Le dernier point est sans doute un cas particulier du précédent : une prédicat simple est une fonction prenant en entrée des nombres de notre arithmétique et renvoyant un booléen.

En géométrie algorithmique, la notion de prédicat est très importante et la robustesse de l'évaluation de ces derniers permet des implémentations robustes d'algorithmes géométriques (la littérature est très dense sur le sujet, voir par exemple [Pio99]). Un prédicat très simple et à la base de beaucoup d'algorithmes est le prédicat d'orientation de trois points dans le plan.

L'objet ici n'est pas de présenter de manière exhaustive toutes les solutions à ce problème de représentation. Nous nous intéresserons à des calculs et prédicats certifiés, dans le sens où nous gardons et propageons les incertitudes tout au long des calculs par l'usage de l'arithmétique d'intervalles dont nous donnons une interprétation en géométrie discrète.

4.3.1 Éléments de base

Supposons le codage des nombres en double précision défini dans la norme IEEE 754, très utilisée dans les calculs numériques : sur 64 bits codant un *double*, nous avons 1 bit de signe (s), 11 bits d'exposant (e) et 52 bits de mantisse (m). Le nombre a pour valeur $s \cdot m \cdot 2^e$. Bien évidemment, tous les nombres réels ne peuvent être représentés, on note donc \mathbb{F} les nombres représentables (auxquels nous pouvons ajouter $\{+\infty, -\infty, NaN\}$). Par conséquence de la construction des doubles, \mathbb{F} n'est pas uniforme sur \mathbb{R} (les petits nombres sont plus finement représentés que les grands).

Si nous considérons un nombre réel x , si x est représentable (*i.e.* $x \in \mathbb{F}$), le double associé est $\underline{x} = x$. Sinon, x est entre deux nombres représentables successifs notés \bar{x} et \underline{x} . Par la suite, nous noterons $\bar{x} = \uparrow x \uparrow$ et $\underline{x} = \downarrow x \downarrow$ les opérateurs permettant d'atteindre les plus proches flottants représentables de x . Pour terminer avec la norme IEEE 754, l'utilisateur choisit un *mode d'arrondi* selon l'opérateur $\uparrow \cdot \uparrow$ ou $\downarrow \cdot \downarrow$ privilégié dans la redéfinition des opérateurs arithmétiques classiques. Ainsi, le résultat d'un calcul sur les nombres représentables est un nombre représentable.

En arithmétique d'intervalles, l'objectif est de propager l'incertitude que l'on a sur la représentation d'un réel tout au long du calcul [Moo66, Moo79]. Tout réel x est représenté par un intervalle X :

$$X = [\bar{x}, \underline{x}]$$

Nous noterons $\mathbb{I}_{\mathbb{F}}$ l'espace engendré par les intervalles sur les nombres représentables \mathbb{F} . Pour simplifier l'écriture, nous notons simplement \mathbb{I} lorsque l'espace *support* des intervalles est, soit implicite par le contexte, soit que la validité de l'énoncé ne dépend pas de celui-ci. En dimension supérieure, \mathbb{I}^n est le produit cartésien $\mathbb{I} \times \dots \times \mathbb{I}$ de n espaces.

Nous pouvons dériver des opérateurs tout en assurant la propriété d'inclusion d'un opérateur sur les intervalles :

PROPRIÉTÉ 4.1 (Inclusion) Soit $f : \mathbb{R} \rightarrow \mathbb{R}$, notons $\square f : \mathbb{I} \rightarrow \mathbb{I}$ la fonction associée à f pour les intervalles. Pour tout $x \in X$, $\square f$ vérifiera la propriété :

$$f(x) = y \Rightarrow y \in \square f(X)$$

Ainsi, nous pouvons définir :

$$\begin{aligned} X \oplus Y &= [\downarrow \underline{x} + \underline{y} \downarrow, \uparrow \bar{x} + \bar{y} \uparrow] \\ X \ominus Y &= [\downarrow \underline{x} - \bar{y} \downarrow, \uparrow \bar{x} - \underline{y} \uparrow] \\ X \odot Y &= [\min(\downarrow \underline{x} \cdot \underline{y} \downarrow, \downarrow \underline{x} \cdot \bar{y} \downarrow, \downarrow \bar{x} \cdot \underline{y} \downarrow, \downarrow \bar{x} \cdot \bar{y} \downarrow), \\ &\quad \max(\uparrow \underline{x} \cdot \underline{y} \uparrow, \uparrow \underline{x} \cdot \bar{y} \uparrow, \uparrow \bar{x} \cdot \underline{y} \uparrow, \uparrow \bar{x} \cdot \bar{y} \uparrow)] \end{aligned}$$

Remarquons l'usage des opérateurs $\uparrow \cdot \uparrow$ et $\downarrow \cdot \downarrow$ permettant de construire \mathbb{I} sur \mathbb{F} , et qu'ainsi, les opérations min et max se font de manière exacte sur \mathbb{F} . Notons aussi que, dans le cas de la multiplication, l'implémentation directe serait inefficace (8 multiplications), en considérant les signes des bornes, nous pouvons obtenir une implémentation plus efficace [Moo79]. Selon ces définitions, l'addition et la multiplication sont associatives et commutatives mais nous n'avons pas la distributivité, en effet :

$$X \odot (Y \oplus Z) \subset (X \odot Y) \oplus (X \odot Z).$$

A partir de ces opérateurs (que nous pouvons compléter), diverses *extensions* sont possibles tout en maintenant la propriété d'inclusion. La plus simple est l'extension naturelle : si $f = g \circ h$ alors $\square f = \square g \circ \square h$. Il est clair qu'il existe une infinité d'extensions possibles de fonctions réelles. Une propriété essentielle, en plus de la propriété d'inclusion, est la propriété de monotonicité pour l'inclusion d'intervalles : si $X \subset Y$, alors nous devons avoir

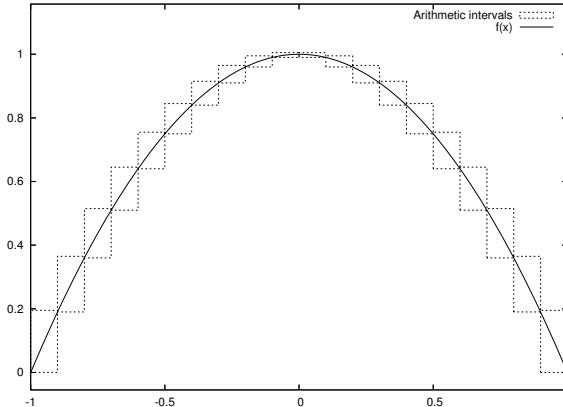


FIGURE 4.7 – Analyse par arithmétique d'intervalles \mathbb{I}_F de la fonction $f(x) = (x+1)(x-1)$ sur $[-1, 1]$ par des intervalles $\{X_i\}_{i=1 \dots 20}$ de taille $1/10$.

$\square f(X) \subset \square f(Y)$. De cette propriété, nous pouvons dériver une notion de convergence très utile lorsque nous construisons un solveur en arithmétique d'intervalles (voir section 4.3.2).

Une autre propriété de l'extension souhaitable mais plutôt difficile à atteindre est l'optimalité (ou minimalité). Si cette dernière est valide, alors, pour tout intervalle X , il n'existe pas d'intervalle $Y \subset \square f(X)$ pour lequel la propriété d'inclusion reste valide ($\forall x \in X, f(x) \in Y$).

Les opérateurs \oplus , \ominus et \odot sont monotones par inclusion et optimaux. L'extension naturelle est aussi monotone mais non optimale [Moo66, Moo79]. En effet, soit $f(x, y) = x * y - x$, son extension naturelle est $\square f(X, Y) = X \odot Y \ominus X$, or pour $X = [-2, 2]$ et $Y = [-1, 5]$, $F(X, Y) = [-12, 12]$. Cependant $[-8, 8]$ vérifie quand même la propriété d'inclusion. $\square f$ n'est donc pas optimale. Cet exemple nous permet aussi d'illustrer la problématique de réécriture syntaxique et symbolique des expressions pour optimiser et contrôler la propagation des erreurs dans les calculs : si nous notons $f(x, y) = x * (y + 1)$, son extension naturelle $\square f(X, Y) = X \odot (Y \oplus [1, 1])$ est elle optimale. Nous pouvons formaliser un peu plus ce fait en reformulant le théorème de MOORE [Moo66] (sous l'hypothèse que les opérateurs de base soient optimaux) :

PROPRIÉTÉ 4.2 ([Moo66]) Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ et $\square f : \mathbb{I}^n \rightarrow \mathbb{I}$ son extension naturelle. Si toutes les variables n'apparaissent qu'une seule fois dans l'expression arithmétique de f , alors $\square f$ est optimale.

D'autres extensions existent, par exemple, si nous cherchons à calculer l'extension de f sachant $g = \nabla f$, nous pouvons utiliser une version sur intervalle du théorème de la valeur moyenne :

$$\square f(X) = f(\text{Mid}(X)) \oplus \square g(X) \odot (X \ominus \text{Mid}(X))$$

$\text{Mid}(X)$ étant la valeur milieu de l'intervalle X . On parle alors d'*extension centrée* [Moo66, RR84]. Cette extension correspond au développement de TAYLOR d'ordre 1 et des généralisations aux ordres supérieurs sont possibles. L'intérêt de ces extensions est que même si l'optimalité ne peut être prouvée, les intervalles sont généralement plus précis qu'avec une extension naturelle.

La figure 4.7 présente une analyse d'intervalles d'une courbe simple. Notons que si la propriété d'inclusion est toujours vraie tout au long du calcul et qu'elle permet donc de prendre des décisions ou de construire des prédictats exacts, la propagation des incertitudes (c'est-à-dire la *largeur* des intervalles) peut être problématique. Pour illustrer cela, l'exemple classique est le suivant (exemple de RUMP) : cherchons à évaluer $f(x, y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + \frac{x}{2y}$, pour $x = 77617$ et $y = 33096$,

sachant que le résultat *exact* est pour les premières décimales, de -0.827396 . Sur certaines arithmétiques (ex. OpenOffice Calc ou Excel), nous obtenons $5.64 \cdot 10^{40}$. Avec une arithmétique d'intervalles basée sur les doubles², nous obtenons $[-5.90296 \cdot 10^{21}, 4.72237 \cdot 10^{21}]$. donc, bien que l'incertitude de l'intervalle soit très grande, il n'en est pas moins vrai que ce dernier contient le résultat exact.

4.3.2 Solveur en arithmétique d'intervalles

Un solveur en arithmétique d'intervalles consiste, de manière générale, à résoudre un système f sur les réels $\mathbb{R}^n \rightarrow \mathbb{R}$ sous forme implicite, c'est-à-dire que nous cherchons les valeurs $\{(x_1, \dots, x_n)\}$ pour lesquels $f(x_1, \dots, x_n) = 0$. Pour cela, l'idée est d'utiliser l'extension $\square f$, et de rechercher les lieux de l'espace, c'est-à-dire les hyper-intervalles $\{X_i\}$ de dimension n tels que $0 \in \square f(X_i)$.

L'idée est de procéder par une recherche dichotomique (voir algorithme 1) : en partant d'un hyper-intervalle initial $X \in \mathbb{I}^n$, nous allons subdiviser récursivement ce domaine en rejetant tous les sous-domaines ne contenant pas la solution. L'arrêt se fait par l'évaluation d'une fonction $\square A(X)$ qui, par exemple, retourne un intervalle contenant 1 tant que l'intervalle X est d'épaisseur supérieure à une constante ϵ (l'épaisseur d'un hyper-intervalle étant la plus grande largeur de ses intervalles mono-dimensionnels). Ce paramètre règle la finesse de l'encadrement des solutions. En contrepartie, la récurrence est d'autant plus longue que l'épaisseur souhaitée est petite.

Sur ce schéma général, de nombreux choix ou optimisations sont possibles comme par exemple le choix entre une liste ou une file pour L (exploration en profondeur ou en largeur d'abord), le choix de la variable lors de la bisection,...

Algorithme 1 : Solveur naïf en arithmétique d'intervalles [Sny92]

Données : l'extension $\square f$, un hyper-intervalle initial X et une fonction d'arrêt $\square A$

Résultat : Une liste S d'hyper-intervalles $\{X_i\}$ tels que $0 \in \square f(X_i)$

Placer X dans une liste L ;

tant que L n'est pas vide faire

Extraire un hyper-intervalle X' de L ;
Evaluation de $\square f(X')$
si $0 \in \square f(X')$ alors
si $1 \in \square A(X)$ alors
Bisection de l'hyper-intervalle X selon une variable x_k en deux sous-hyper-intervalles X' et X'' ;
Ajout de X' à L ;
Ajout de X'' à L
sinon
Ajout de X à S

En informatique graphique, ce solveur peut-être utilisé pour le tracé de courbes et de surfaces implicites [Sny92]. Néanmoins, des solveurs plus élaborés sont étudiés dans la communauté de programmation par contraintes dans le cas de la résolution de contraintes non-linéaires sur les réels (voir par exemple [BMAVH94, RS97]).

Etant donné que nous pouvons définir, pour cette arithmétique, des opérateurs ensemblistes (intersection, union,...) ou encore booléens, l'extension $\square f$ peut être complexe. Par exemple, nous pouvons ainsi modéliser l'union ou encore l'intersection de formes implicites et utiliser le solveur précédent pour avoir une approximation de la solution.

2. <http://www.losderover.be/old/Easyval/>

4.3.3 Lien avec un modèle analytique discret de supercouverture

D'un point de vue conceptuel, la propriété d'inclusion dans les modèles analytiques discrets (équation 4.4) est à mettre en relation avec la propriété d'inclusion, fondement de l'arithmétique d'intervalles : dans les deux modèles, nous avons toujours une inclusion de l'objet continu sous-jacent. Là où le modèle analytique discret propose une construction géométrique des objets (intersection avec un élément structurant, distance à l'objet, ...), les intervalles proposent une analyse arithmétique, voire algébrique.

Pour avancer un peu sur les liens entre ces modèles, nous allons considérer l'arithmétique $\mathbb{I}_{\mathbb{Z} + \frac{1}{2}}$ définissant les intervalles sur les demi-entiers (nombres $k + \frac{1}{2}$, pour $k \in \mathbb{Z}$). Ce décalage de $\frac{1}{2}$ nous permet de centrer les intervalles sur les points discrets. Les intervalles se construisent de la manière suivante :

$$\text{si } k \in \mathbb{Z}, \quad K = \left[k - \frac{1}{2}, k + \frac{1}{2} \right].$$

Lorsqu'un nombre est donné sur les réels ou lors de la définition d'un opérateur arithmétique, nous définissons les opérateurs :

$$\begin{aligned}\uparrow x \uparrow &= \left[x + \frac{1}{2} \right] - \frac{1}{2}, \\ \downarrow x \downarrow &= \left[x + \frac{1}{2} \right] - \frac{1}{2}.\end{aligned}$$

De manière simple, nous pouvons dériver les opérateurs arithmétiques sur cet espace $\mathbb{I}_{\mathbb{Z} + \frac{1}{2}}$. Prenons l'exemple tracé sur la figure 4.8-(a) de la fonction $f(x) = \frac{1}{2}x + 1$, l'analyse de $\square f$ pour les intervalles de taille 1 centrés sur les points discrets nous donne exactement le même résultat que le tracé $\mathbb{S}(f)$, modulo le fait que deux pixels alignés verticalement dans $\mathbb{S}(f)$ nous donnent pour $\square f$ un simple intervalle de hauteur 2.

Avant de formaliser les interactions entre ces modèles, il faut cependant régler le problème des *bulles* (voir section 4.2.2). En effet, dans notre espace support des bornes des intervalles, $\mathbb{Z} + \frac{1}{2}$, si x est représentable (c'est-à-dire sous la forme $k + \frac{1}{2}$ pour $k \in \mathbb{Z}$), il est d'usage dans les bibliothèques implémentant les arithmétiques d'intervalles de considérer l'intervalle de largeur nulle $[x, x]$ (afin de supprimer une incertitude qui se propagerait inutilement dans les calculs). L'inconvénient dans notre mise en relation avec le modèle supercouverture est que lors du tracé sur la figure 4.8-(b) avec la fonction, $f(x) = \frac{1}{3}x$ supprimera de manière intrinsèque toutes les bulles, en violation avec les principes de construction, notamment morphologiques, de $\mathbb{S}(f)$. Nous choisissons donc de contraindre notre arithmétique pour retourner l'intervalle $[x - 1, x + 1]$ si $x \in \{\mathbb{Z} + \frac{1}{2}\}$.

Avec ce choix spécifique, nous obtenons des algorithmes de tracé de fonctions très simples (voir figure 4.9). Remarquons sur la figure 4.9-(a) la présence de bulles (même fonction que pour la figure 4.8-(b)). La figure 4.9-(b) illustre l'utilisation d'opérateurs plus complexes. La figure 4.10 illustre différents tracés pour lesquels l'arithmétique d'intervalles n'est pas optimale.

Nous pouvons maintenant formaliser plus précisément les liens entre la supercouverture et l'analyse par intervalles. Dans le lemme suivant, l'inclusion et l'égalité sont à prendre au sens géométrique : inclusion ou égalité des régions du plan engendrées par l'union de cellules fermées de la grille pour la supercouverture, et l'union des hyper-intervalles 2D pour l'arithmétique d'intervalles.

LEMME 4.1 *Supposons l'arithmétique sur $\mathbb{I}_{\mathbb{Z} + \frac{1}{2}}$ et $f : \mathbb{R} \rightarrow \mathbb{R}$ une application continue,*

4.3. Arithmétique d'intervalles et supercouverture

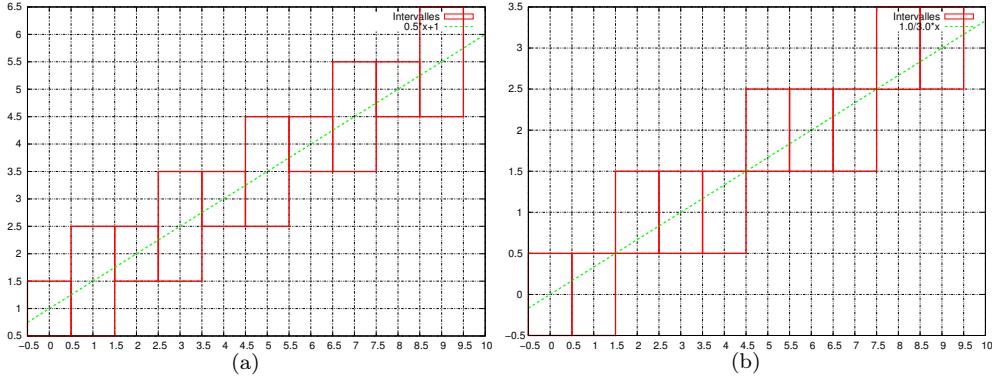


FIGURE 4.8 – Tracé en arithmétique d'intervalles $\mathbb{I}_{\mathbb{Z}+\frac{1}{2}}$ des fonctions $f(x) = \frac{1}{2}x + 1$ (a) et $f(x) = \frac{1}{3}x$ (b).

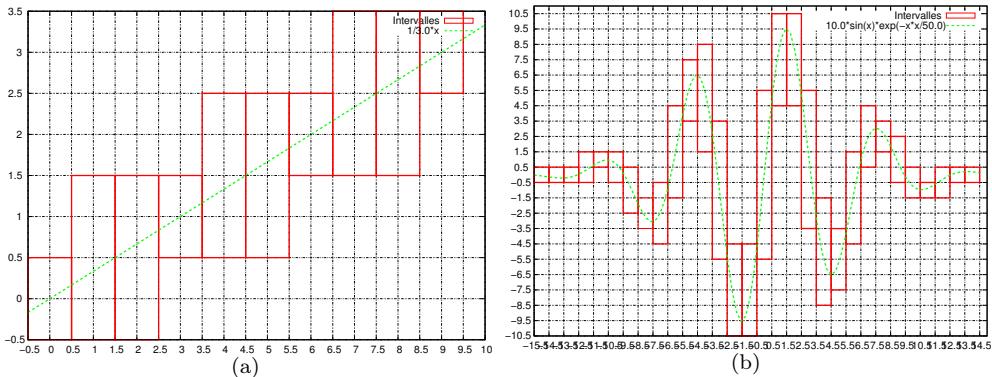


FIGURE 4.9 – Tracé en arithmétique d'intervalles $\mathbb{I}_{\mathbb{Z}+\frac{1}{2}}$ modifiée, des fonctions $f(x) = \frac{1}{3}x$ et $f(x) = 10 \sin(x) \exp^{-x^2/50}$.

*alors*³ :

$$\mathbb{S}(f) \subseteq \bigcup_{k \in \mathbb{Z}} [K \times \square f(K)].$$

Si $\square f$ est optimale sur $\mathbb{I}_{\mathbb{Z} + \frac{1}{2}}$, alors :

$$\mathbb{S}(f) = \bigcup_{k \in \mathbb{Z}} [K \times \square f(K)].$$

Preuve : Considérons un intervalle centré sur l'abscisse k : $K = [k - \frac{1}{2}, k + \frac{1}{2}]$. Nous notons $Y = \square f(K)$. Soit C une cellule de $\mathbb{S}(f)$ centrée sur l'abscisse k . Étant donné que $\mathbb{S}(f)$ est l'ensemble des cellules du pavage régulier intersectant l'objet continu, il existe $x \in K$ tel que $f(x) \cap C \neq \emptyset$. Par ailleurs et par définition de $\mathbb{I}_{\mathbb{Z}+\frac{1}{2}}$, $f(x) \in Y$ mais aussi $[\downarrow f(x), \uparrow f(x)] \subseteq Y$. Par définition du modèle en supercouverture, C étant défini par $[k - \frac{1}{2}, k + \frac{1}{2}] \times [\downarrow f(x), \uparrow f(x)]$, nous concluons que $C \subseteq Y$ et donc $\mathbb{S}(f) \subseteq \square f$.

Pour montrer l'égalité si $\square f$ est optimale, supposons une cellule $C : [k - \frac{1}{2}, k + \frac{1}{2}] \times [l - \frac{1}{2}, l + \frac{1}{2}]$ telle que $C \subset Y$ et telle que $l - \frac{1}{2}$ soit aussi la borne inférieure de Y . Nous

3. K est l'intervalle $\left[k - \frac{1}{2}, k + \frac{1}{2}\right]$ pour $k \in \mathbb{Z}$.

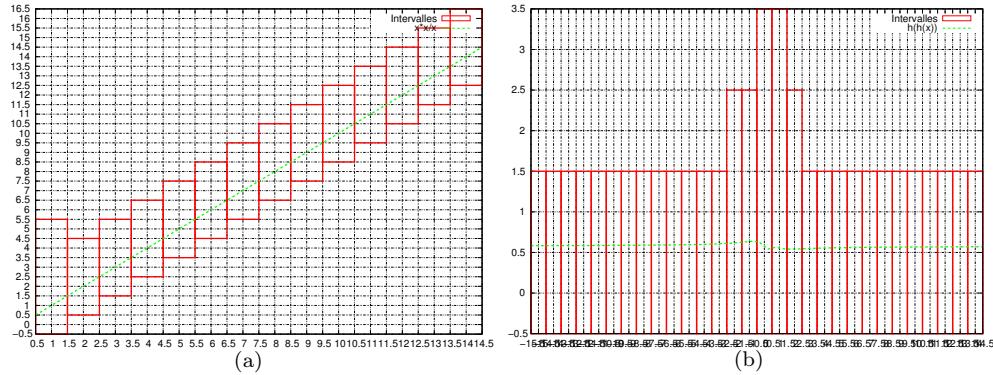


FIGURE 4.10 – Exemple de tracé avec des extensions non optimales : (a) $f(x) = \frac{x^2}{x}$ et (b) $g(x) = h(h(x))$ avec $h(x) = \frac{\sqrt{x^2-x+1/2}}{\sqrt{x^2+1/2}}$.

montrons que si $C \not\subseteq \mathbb{S}(f)$ alors $C \not\subseteq \square f(K)$. Si $C \not\subseteq \mathbb{S}(f)$, alors pour tout $x \in K$, $f(x) \cap C = \emptyset$. Donc il existe un intervalle $Y' = Y/C$ (Y' est donc de la forme $[l + \frac{1}{2}, p]$ avec $p \in \mathbb{Z} + \frac{1}{2}$) tel que pour tout $x \in K$, $f(x) \in Y$, contredisant le fait que $\square f$ soit optimale. Ainsi, $C \not\subseteq \square f(K)$. Un raisonnement similaire peut être donné dans le cas où la borne supérieure de C coïncide avec la borne supérieure de Y . Le dernier cas à traiter est lorsque C n'est pas une extrémité de Y . Dans ce cas et par continuité de f , il n'existe pas de cellules C' au-dessus de C et C'' en-dessous telles que C' et C'' appartiennent tous deux à $\mathbb{S}(f)$ et non C . Ainsi, si $C \not\subseteq \mathbb{S}(f)$ alors nécessairement, ou bien toutes les cellules en-dessous de C ne sont pas dans $\mathbb{S}(f)$, ou alors c'est le cas pour celles au-dessus de C . Nous pouvons donc nous ramener dans le cas de la preuve où nous restreignons Y par une de ces extrémités (voir figure 4.11).

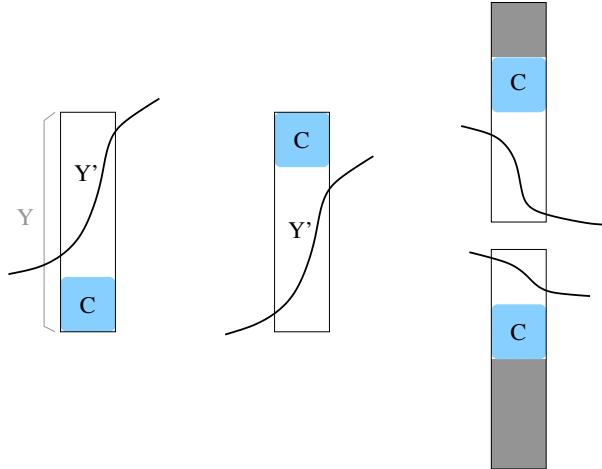


FIGURE 4.11 – Illustration de la preuve du lemme 4.1 : les différents cas de position de C dans Y .

□

Ce lemme très simple nous permet d'ajouter au modèle de supercouverture, en plus des caractérisations géométriques, de morphologie mathématique ou de distance, une caractérisation arithmétique par l'usage d'opérateurs sur les intervalles. Notons que nous pouvons très simplement généraliser ce lemme en dimension supérieure.

La figure 4.10 illustre le cas de tracé avec des extensions non-optimales. Si l'inclusion $\mathbb{S}(f) \subseteq \square f$ est bien évidemment vérifiée, les propagations d'erreurs peuvent produire des encadrements très grands de la solution.

4.4 Conclusion

Dans ce chapitre nous avons présenté les différents modèles de discréétisation qui seront utilisés dans les chapitres suivants. Si tous ces modèles sont classiques en géométrie discrète, nous avons cependant proposé une approche algébrique à l'un de ces modèles par le biais d'une arithmétique d'intervalles spécifique.

Dans le chapitre 6, nous reviendrons sur une utilisation plus générale de l'arithmétique d'intervalles pour le tracé et la reconstruction réversible de fonctions complexes sur des grilles isothétiques irrégulière.

Chapitre 5

Algorithmes de reconnaissance de parties linéaires et de cercles

Sommaire

5.1	Introduction	48
5.2	Définitions des objets élémentaires	48
5.2.1	Droites discrètes	48
5.2.2	Plans et hyperplans discrets	49
5.2.3	Cercles, sphères et hypersphères discrètes	50
5.3	La boîte à outils	51
5.3.1	Programmation linéaire	51
5.3.2	Programmation linéaire entière	52
5.3.3	Polytopes discrets	53
5.3.4	Enveloppe convexe	54
5.3.5	Séparabilité par cercles	54
5.4	Algorithmes de reconnaissance	55
5.4.1	Droites discrètes	55
5.4.2	Plans et hyperplans discrets	55
5.4.3	Préimage et enveloppe convexe d'un hyperplan discret	57
5.4.4	Cercles discrets	61
5.5	Conclusion	64

5.1 Introduction

Pour définir de manière intrinsèque un paradigme de géométrie discrète, il nous faut des objets élémentaires. Le chapitre précédent nous a donné quelques clés pour définir des formes discrètes et donc des points, des droites, des cercles, ... Dans ce chapitre nous revenons sur ces objets élémentaires et sur le problème de leur reconnaissance qui consiste à passer d'une représentation en extension (ensemble de points discrets) à une description en compréhension (droite discrètes, cercle discret,...). Nous aborderons ces problèmes d'un point de vue algorithmique en exploitant de nombreux résultats venant d'arithmétique, de géométrie algorithmique ou de programmation linéaire.

Nous commencerons par définir les objets élémentaires, nous présenterons une série d'outils de différents domaines pour aboutir à une présentation synthétique des différentes solutions algorithmiques.

5.2 Définitions des objets élémentaires

Dans ces paragraphes, nous ne présenterons que les caractérisations d'ordre géométrique des objets élémentaires. En réalité, ces objets fondamentaux sont beaucoup plus riches que ces simples caractéristiques. C'est le cas par exemple de la droite discrète dont les structures périodiques dans le cas rationnel trouvent des échos très forts en combinatoire, en théorie des mots ou encore en arithmétique. Une littérature importante existe faisant le pont entre ces objets et des résultats d'autres domaines. Nous pouvons citer quelques références essayant de faire un point aussi exhaustif que possible sur ces liens pour les droites discrètes [KR04b, KR04a] ou pour les plans discrets [BCK07].

5.2.1 Droites discrètes

Au cours du temps, de nombreuses caractérisations des droites discrètes ont été données depuis les premiers algorithmes de tracé sur écran matriciel [KR04a]. Une façon simple de définir une droite discrète est de considérer la discréttisation d'une droite réelle étant donné un processus de discréttisation. Si cette approche *externe* est possible, nous préférons ici une définition arithmétique proposée par REVEILLÈS [Rev91] :

DÉFINITION 5.1 (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 $a, b \in \mathbb{Z}$, $\mu \in \mathbb{N}^*$, $b \neq 0$ et $\text{pgcd}(|a|, |b|) = 1$), si et seulement si tous les pixels (x, y) de \mathcal{E} vérifient :*

$$\mu \leq ax - by < \mu + \omega$$

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

Le paramètre ω nous permet de régler l'épaisseur de la droite discrète. Ainsi, si $\omega = |a| + |b|$, la courbe construite sera une (1)-courbe (*droite standard*). Si $\omega = \max(|a|, |b|)$, la courbe sera (0)-connexe (*droite naïve*). Pour les cas intermédiaires, la courbe sera soit épaisse, soit déconnectée [Rev91, Deb95] (voir figure 5.1).

Si historiquement cette droite était définie de manière intrinsèque au modèle discret, de nombreuses équivalences existent avec les discréttisés des droites réelles. Ainsi, les droites issues des discréttisations GIQ, BBQ ou OBQ (voir section 4.2.1) sont des droites naïves dont les paramètres sont donnés explicitement [Rev91]. Par la suite, on appelle segment discret un ensemble connexe de pixels d'une droite discrète.

Associé à cette droite arithmétique, nous pouvons caractériser l'ensemble des droites réelles liées à un segment discret. Pour cela, revenons à un processus basé sur la discréttisation

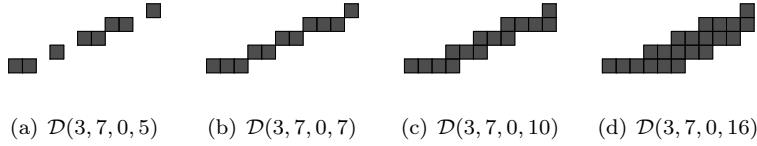


FIGURE 5.1 – Connexité des droites discrètes en fonction de ω : (a) segment déconnecté, (b) segment naïf, (c) segment standard et (d) segment épais.

OBQ de la droite $y = \alpha x + \beta$, l'ensemble des points discrets considérés est :

$$S = \{(x, y) \in \mathbb{Z}^2, \quad 0 \leq \alpha x + \beta - y < 1\} \quad (5.1)$$

Ainsi l'ensemble des droites réelles de la forme $y = \alpha x + \beta$ dont la discréétisation OBQ contient un segment discret S donné est :

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

Cela revient à considérer l'espace des solutions du problème de programmation linéaire engendré par les contraintes de l'équation (5.1) (voir section 5.3.1).

Dans l'espace des paramètres (α, β) , \bar{S} est un polygone convexe et est appelé *préimage* de S . Tout point (α, β) dans \bar{S} définit une droite dont la discréétisation OBQ contient l'ensemble S (voir figure 5.2). Ce domaine sera utilisé dans les algorithmes de reconnaissance car de nombreuses propriétés arithmétiques des droites discrètes vont se retrouver comme propriété géométrique sur la préimage. Par exemple, si S est (0) -connexe, \bar{S} est un polygone ayant au plus 4 sommets [DS84, Mci85, LB93]. Notons que les coordonnées des sommets de \bar{S} peuvent être calculées explicitement à partir des paramètres arithmétiques a, b et μ du segment discret [Coe02].

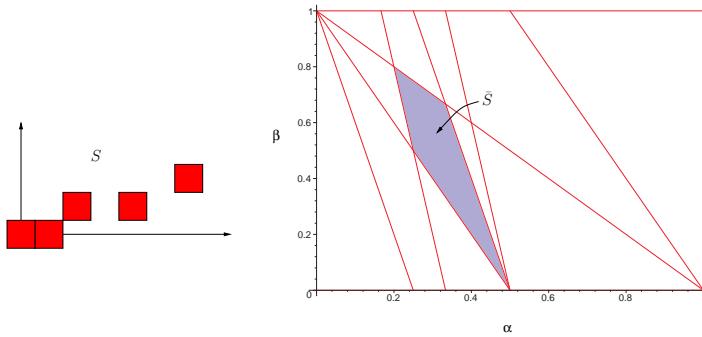


FIGURE 5.2 – Préimage d'un ensemble de points discrets S .

5.2.2 Plans et hyperplans discrets

Même si la littérature est dense spécifiquement sur les plans discrets, nous allons définir cet objet dans le même temps que les hyperplans discrets. Trois grandes définitions existent pour définir cet objet géométrique dans \mathbb{Z}^n . Si ces trois définitions comportent des similitudes et si elles sont parfois équivalentes, les processus de discréétisation liés à ces définitions peuvent varier. Nous notons Γ un hyperplan euclidien donné par $\gamma_0 + \sum_{i=1}^n \gamma_i x_i = 0$ avec

$\{\gamma_i\} \in \mathbb{R}$, $|\gamma_i| \leq |\gamma_n|$ pour $1 \leq i < n$ et $|\gamma_n| > 0$ (l'axe x_n est appelé axe principal de l'hyperplan, voir plus loin). Dans les définitions suivantes, nous gardons les termes anglais pour éviter toute ambiguïté.

DÉFINITION 5.2 (Digital hyperplane [ST91]) $S \subset \mathbb{Z}^n$ est un digital hyperplane si et seulement si il existe un hyperplan euclidien Γ tel que S soit l'image digitale de Γ .

La notion d'*image digitale* correspond à un processus de discréétisation spécifique qui construit les points discrets dont la distance verticale (L_∞) sur la n -ième coordonnée entre le point discret et le plan est inférieure à $\frac{1}{2}$.

DÉFINITION 5.3 (Digital flatness [Vee93]) $S \subset \mathbb{Z}^n$ est flat si et seulement s'il existe $n+1$ nombres réels $\gamma_0, \dots, \gamma_n$ tels que :

1. $\max\{|\gamma_0|, \dots, |\gamma_n|\} = 1$;
2. tout point $P = (P_1, \dots, P_n) \in S$ vérifie la condition

$$-\frac{1}{2} < \gamma_0 + \sum_{i=1}^n \gamma_i P_i \leq \frac{1}{2}. \quad (5.2)$$

Dans [Vee93] VEELAERT montra qu'un *digital hyperplan* de la définition 5.2 est *flat*. Enfin nous avons une définition à mettre en parallèle avec la notion de droite discrète arithmétique :

DÉFINITION 5.4 (Discrete analytic hyperplane [AAS97]) S est un morceau de discrete analytic hyperplane P ayant pour coefficients $(a_1, a_2, \dots, a_n, b) \in \mathbb{Z}^{n+1}$, $\text{pgcd}(a_1, \dots, a_n) = 1$, et pour épaisseur $\omega \in \mathbb{N}^*$ si :

$$S \subset P(a_1, a_2, \dots, a_n, b) = \{(x_1, x_2, \dots, x_n) \in \mathbb{Z}^n \mid 0 \leq b + \sum_{i=1}^n a_i x_i < \omega\} \quad (5.3)$$

Comme pour les droites, cette définition intègre une notion d'épaisseur. Tout comme en dimension 2, si $\omega = \max_{i=1..n} \{|a_i|\}$, elle définit un hyperplan discret naïf [AAS97]. Par la suite, c'est cet objet que nous considérerons dans les algorithmes de reconnaissance.

Si les coefficients du plan euclidien Γ dans la définition 5.2 sont rationnels, alors un *digital hyperplane* est aussi un *digital analytic hyperplane*.

Nous concluons ces définitions par la notion d'axe principal et de base d'un hyperplan discret.

LEMME 5.1 (Axe principal et base d'un hyperplan discret) Soit S un digital hyperplane ou un hyperplan discret naïf. Il existe alors un axe principal x_j tel que la projection S' de S selon l'axe x_j sur le plan $x_j = 0$ soit une bijection. L'ensemble de points discrets de dimension $(n-1)$ S' est appelé la base de l'hyperplan.

Ce lemme est issu des différentes définitions et est généralisé dans [BFJP07]. Par exemple pour la définition 5.2 l'axe principal est x_n , dans la définition 5.3, l'axe est x_j si $|\gamma_j| = 1$. Enfin, pour les hyperplans discrets naïfs, l'axe principal est x_j si $\omega = |\gamma_j|$ (voir figure 5.3).

De la même manière que pour les droites discrètes, nous pouvons définir la préimage comme étant l'ensemble des hyperplans euclidiens dont le discréétisé contient un ensemble S de points discrets. Nous reviendrons dans la section 5.4 sur cet objet.

5.2.3 Cercles, sphères et hypersphères discrètes

Toujours en fonction d'un modèle de discréétisation, plusieurs définitions existent pour les cercles discrets ou pour les sphères et hypersphères. Une définition générale a cependant été donnée par ANDRÈS et JACOB :

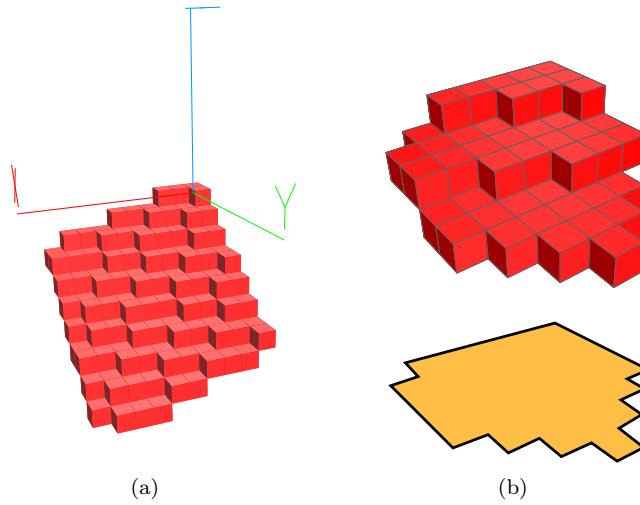


FIGURE 5.3 – Morceau de plan discret naïf de paramètres $P(6, 13, 17, 17)$ (a) et illustration de la base 2D associé à un morceau de plan.

DÉFINITION 5.5 (Hypersphère discrète analytique [AJ97]) Une hypersphère discrète analytique de dimension n , de centre $C \in \mathbb{R}^n$, de rayon $r \in \mathbb{R}$ et d'épaisseur $\omega \in \mathbb{R}$ est l'ensemble des points discrets :

$$H_n(C, r, \omega) = \left\{ P \in \mathbb{Z}^n, \left(r - \frac{\omega}{2}\right)^2 \leq \sum_{i=1}^n (C_i - P_i)^2 < \left(r + \frac{\omega}{2}\right)^2 \right\}$$

Dans [AJ97], les auteurs discutent du paramètre d'épaisseur en fonction de la connexité souhaitée pour l'ensemble discret obtenu. D'autres résultats de connectivité ainsi qu'une formulation analytique du cercle de BRESENHAM [Bre77], très classique en informatique graphique, sont disponibles dans [FTJ06, FT06]. Dans le cas du cercle de BRESENHAM, si une telle caractérisation existe, elle met en place une épaisseur non constante et ne rentre donc pas entièrement dans le cadre de la définition 5.5.

De cette définition analytique, nous retenons dans les algorithmes de reconnaissance de la section 5.4 le principe suivant (pour la dimension 2) : un ensemble S de points discrets est un disque discret s'il existe un cercle séparant S de son complémentaire \bar{S} . Nous nous ramènerons ainsi à un problème de séparabilité.

5.3 La boîte à outils

Dans ces paragraphes, nous présentons différents outils de domaines variés permettant d'offrir des solutions algorithmiques efficaces à la reconnaissance des objets élémentaires.

5.3.1 Programmation linéaire

Un problème de programmation par contrainte consiste à maximiser une fonction objectif linéaire étant donné un ensemble de contraintes linéaires. Le système de contraintes peut se définir de manière générale par :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2 \\ \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m \end{cases}$$

ou plus simplement par une représentation matricielle :

$$Ax \leq b$$

avec $A = (a_{ij}) \in \mathbb{R}^{m \times n}$ et $b = (b_i) \in \mathbb{R}^m$. La fonction objectif est donnée de manière générale sous la forme cx avec $c = (c_j) \in \mathbb{R}^n$.

On cherche donc à résoudre :

$$\begin{aligned} & \max cx \\ & Ax \leq b \\ & x \in \mathbb{R}^n \end{aligned}$$

De nombreux algorithmes existent pour résoudre ce système (noté LP) [Sch86]. D'un point de vue théorique, nous avons le résultat suivant :

THEORÈME 5.1 ([Meg84a]) *Étant donné un problème LP donné avec m contraintes linéaires dans \mathbb{R}^n où n est fixe, il existe alors un algorithme permettant de résoudre le problème en $O(m)$.*

Si ce résultat est très fort d'un point de vue théorique, l'algorithme suggéré par MEGIDDO n'est implantable que pour de faibles dimensions (2, voire 3). Notons aussi que la constante dans la borne linéaire est très grande (doublement exponentielle en la dimension). En pratique, on préfère utiliser des techniques parfois exponentielles en temps (comme le Simplex) mais très rapides en pratique, ou encore des approches *randomisées* (voir par exemple [Gol95]).

5.3.2 Programmation linéaire entière

Un problème de programmation linéaire entière est un problème linéaire dans lequel les contraintes sont données par des coefficients entiers et dont on cherche des solutions entières [Sch86]. Le problème s'écrit donc :

$$\begin{aligned} & \max cx \\ & Ax \leq b \\ & x \in \mathbb{Z}^n \end{aligned}$$

sachant $A = (a_{ij}) \in \mathbb{Z}^{m \times n}$, $b = (b_i) \in \mathbb{Z}^m$ et $c = (c_j) \in \mathbb{Z}^n$.

Décider si une solution $x \in \mathbb{Z}^n$ existe est un problème NP-complet [GJ79]. Cependant, de nombreux travaux visent à caractériser cet ensemble de point discret s'il existe [Sch86, BHL92, Zol00, Bár02].

Le système de contraintes A et b définit un polytope \mathcal{P} de \mathbb{R}^n qui correspond au domaine dans lequel il faut chercher les points discrets solutions. Considérons P l'enveloppe convexe de $\mathcal{P} \cap \mathbb{Z}^n$. C'est ce polytope discret que nous allons caractériser. On note $N(A, b)$ l'ensemble des sommets de P et nous nous intéressons à $|N(A, b)|$.

THEORÈME 5.2 ([Zol00]) *Soient $A = (a_{ij}) \in \mathbb{Z}^{m \times n}$, $b = (b_i) \in \mathbb{Z}^m$, $c = (c_j) \in \mathbb{Z}^n$ et $\alpha = \max\{|a_{ij}|, i = 1, \dots, m, j = 1, \dots, n\}$. Nous avons,*

$$|N(A, b)| \leq c_n m^{\lfloor n/2 \rfloor} \log^{n-1}(1 + \alpha) \quad (5.4)$$

où c_n est une quantité dépendant uniquement de la dimension n .

Ce résultat nous servira à borner l'enveloppe convexe d'un morceau de plan discret et de sa préimage dans la section 5.4.2.

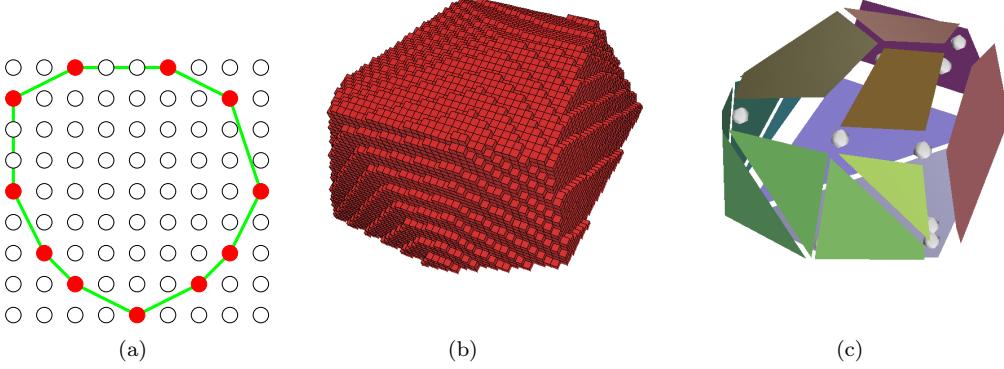


FIGURE 5.4 – (a) Exemple de polygone convexe discret inclus. Les figures (b) et (c) illustrent la combinatoire entre l'objet discret (ici 3D) et les sommets de son enveloppe convexe (c).

5.3.3 Polytopes discrets

Dans la section précédente, nous avons déjà présenté des résultats sur les polytopes discrets lorsque ceux-ci sont définis comme l'espace des solutions d'un problème de programmation linéaire entière. Ci-dessous, nous caractérisons le nombre de sommets d'un polytope discret si celui-ci est donné par la discrétisation (au sens de GAUSS) d'un polytope continu ayant certaines propriétés.

Soit P un polytope discret de dimension n non-vide. Nous notons $f_k(P)$ le nombre de facettes de dimension k de P ($0 \leq k < n$, par exemple f_0 est le nombre de sommets de P). Nous avons alors le résultat suivant :

THEORÈME 5.3 ([BL98])

$$f_k \leq c_n (\text{Vol } P)^{\frac{n-1}{n+1}} \quad (5.5)$$

où c_n ne dépend que de la dimension n et $(\text{Vol } P)$ est le volume de P .

En dimension 2, ce résultat est équivalent à un résultat d'ACKETA et ŽUNIĆ qui permet de borner le nombre d'arêtes $e(N)$ d'un polygone convexe discret contenu dans une fenêtre $N \times N$ [BB91, AŽ95] :

$$e(N) = \frac{12}{(4\pi^2)^{1/3}} N^{2/3} + O(N^{1/3} \log(N)). \quad (5.6)$$

Le théorème 5.3 peut même être généralisé en considérant non plus un polytope discret P , mais la discrétisation de GAUSS d'une forme continue ayant certaines propriétés :

THEORÈME 5.4 ([BL98]) Soit $K \in C(D)$ où $C(D)$ est la famille des formes convexes ayant un bord C^2 et ayant un rayon de courbure en tout point et dans toutes les directions entre $1/D$ et D , pour $D \geq 1$. Notons $\bar{K} = \text{conv}(K \cap \mathbb{Z}^n)$. Si le diamètre δ de K est suffisamment grand, alors pour tout $n \geq 2$, il existe deux constantes $c_1(n)$ et $c_2(n)$ telles que pour tout $k \in \{0, 1, \dots, n-1\}$,

$$c_n \delta^{n \frac{n-1}{n+1}} \leq f_k(\bar{K}) \leq c'_n \delta^{n \frac{n-1}{n+1}}. \quad (5.7)$$

Ces résultats sont fondamentaux pour de nombreux aspects de la reconnaissance de formes discrètes élémentaires. C'est le cas pour les plans discrets ou les cercles discrets. Ils sont de plus à la base de différentes preuves de convergence asymptotique d'estimateurs géométriques comme la longueur ou les normales de contours discrets 2D [dV07].

5.3.4 Enveloppe convexe

A de nombreux endroits des algorithmes de la section 5.4, ces derniers nécessitent des constructions d'enveloppes convexes [PS85, BY95, GO97]. Dans le tableau 5.1, nous reprenons quelques résultats de complexité. Dans ce tableau, m correspond au nombre de points considérés, n la dimension et h la taille du résultat. Les algorithmes incorporant cette quantité h dans leur mesure de complexité sont dits *output sensitive*, c'est-à-dire que le coût est lié à la taille du résultat. Les algorithmes incrémentaux nous permettent d'insérer les points un à un et de maintenir à chaque instant une enveloppe convexe des points traités. On peut vouloir quantifier le coût algorithmique de l'ajout d'un point. Par algorithme dynamique, nous entendons un algorithme dans lequel nous pouvons ajouter et supprimer des points.

Nous avons aussi ajouté au tableau deux résultats dans le cas où l'ensemble de points en entrée est à support discret. Dans ce cas, le calcul d'enveloppe convexe d'une courbe discrète est linéaire en son nombre de points [Vos93]. Notons que si la courbe discrète est vue comme une chaîne, les résultats de [Mel87] ou [Buz07] peuvent aussi être utilisés. L'algorithme de HAR-PELED [Har98] est cependant à mettre en avant : cet algorithme est *output sensitive* et son coût est indépendant (dans une certaine mesure) de la cardinalité de X , dépendant seulement de son diamètre $\delta(X)$. Notons cependant que les hypothèses sur la classe d'objets X permettant d'atteindre cette borne sont fortes.

Objets	Complexité au pire cas	Références
Points 2D	$O(m \log m)$	GRAHAM [Gra72]
Points 2D	$O(mh)$	JARVIS [Jar73]
Points 2D	$O(m \log h)$	CHAN [Cha96]
Points 3D	$O(m \log h)$	CHAN [Cha96]
Points dD	$O(m \log m + m^{\lfloor n/2 \rfloor})$	CHAZELLE [Cha93]
Chaîne simple 2D	$O(m)$ incrémental $O(1)$	MELKMAN [Mel87]
Chaîne simple 2D	$O(m)$ dynamique $O(1)$	BUZER [Buz07]
Courbe discrète	$O(m)$	VOSS [Vos93]
Objet discret convexe X	$O(h \log \delta(X))$	HAR-PELED [Har98]

TABLE 5.1 – Enveloppes convexes et principales références.

5.3.5 Séparabilité par cercles

Pour terminer avec les outils dont nous aurons besoin, nous présentons quelques résultats concernant le problème de séparation de deux ensembles par des cercles en géométrie algorithmique.

Le problème peut-être exposé de la façon suivante : étant donnés deux ensembles S et T , comment décider s'il existe un cercle contenant tous les points de S et ne contenant aucun point de T ? Autrement dit, est-ce que S et T sont séparables par un cercle?

Si nous notons $m = |S| + |T|$, O'Rourke *et al.* [OKM86] ont proposé une réécriture du problème en termes de programmation linéaire en dimension 3 de m contraintes. En utilisant les résultats de MEGIDDO (voir section 5.3.1), nous obtenons les résultats présentés dans le tableau 5.2. Notons que si d'un point de vue théorique, la résolution d'un problème linéaire en dimension 3 est linéaire au nombre de contraintes, l'implémentation de cet algorithme est très complexe.

Si nous nous restreignons à des outils en dimension 2, nous pouvons considérer un test de séparabilité basé sur une notion de cellule de VORONOÏ généralisée construite en considérant tous les couples (s, t) avec $s \in S$ et $t \in T$ [CGRT04]. Nous obtenons ainsi des coûts supérieurs mais l'implémentation de ces algorithmes est réalisable (voir tableau 5.2).

Caractéristique	Complexité au pire cas	Références
Test de séparabilité (LP dimension 3)	$O(m)$	[OKM86]
Extraction du cercle séparant de rayon minimal	$O(m)$	[OKM86]
Extraction de tous les cercles séparants	$O(m \log m)$	[OKM86]
Test séparabilité (LP dimension 2)	$O(S \cdot T)$	[PS85, CGRT04]
Extraction de tous les cercles séparants	$O(S \cdot T \log(S \cdot T))$ incrémental	[PS85, CGRT04]

TABLE 5.2 – Principales références pour la séparabilité de deux ensembles par un cercle.

5.4 Algorithmes de reconnaissance

Nous allons maintenant combiner tous les outils précédents pour synthétiser les différentes approches pour la conception des algorithmes de reconnaissance et pour présenter leur coût algorithmique.

Quel que soit l'objet élémentaire reconnu et étant donné qu'il existera toujours un nombre infini de représentants euclidiens associés à un ensemble fini de points discrets, deux grandes classes d'algorithmes peuvent être définies : la première contient les algorithmes de détection qui ont soit une réponse binaire au problème de décision associé, soit retournent un unique représentant euclidien. La seconde classe reprend les algorithmes basés sur une notion de préimage nous retourneront toutes les solutions sous forme généralement d'un polytope dans un certain espace des paramètres.

5.4.1 Droites discrètes

Concernant les algorithmes de reconnaissance de droite discrète, le problème a été très étudié et des solutions efficaces existent. Le tableau 5.3 synthétise quelques-unes de ces solutions. Rappelons que les paramètres issus des algorithmes de reconnaissance dits “arithmétiques” décrivent complètement la préimage associée aux droites discrètes.

D'un point de vue pratique, les algorithmes de reconnaissance sont assez simples et ne nécessitent pas d'outils très complexes.

Contrainte	Caractéristique	Complexité au pire cas	Références
Courbe connexe	Calcul de bande minimale	$O(m)$ incrémental $O(1)$	[Kov90]
	Reco. arithmétique	$O(m)$ incrémental $O(1)$	[DR95, Deb95]
	Reco. arithmétique	$O(m)$ dynamique $O(1)$	[dV07]
	Préimage	$O(m)$ incrémental $O(1)$	[DS91, LB93]
Non connexe mais avec ordre	Préimage	$O(m)$ incrémental	[O'R81]
Non connexe	Test linéarité LP dimension 2	$O(m)$ incrémental	[Meg84c, Buz03]
	Preimage LP dimension 2	$O(m \log m)$ incrémental	[PS85]

TABLE 5.3 – Algorithmes de reconnaissance de droite discrète.

5.4.2 Plans et hyperplans discrets

Concernant les plans et hyperplans discrets, les problèmes sont plus complexes. Intuitivement, nous avons peu d'algorithmes exploitant les propriétés arithmétiques et de périodicités des plans discrets comme c'était le cas en dimension 2. Cependant, nous avons des techniques

fondées sur des outils classiques de géométrie algorithmique mais dont les complexités sont modulées par le caractère discret des objets en exploitant la combinatoire spécifique des polytopes discrets.

Pour la construction des algorithmes de reconnaissance, nous allons utiliser différents résultats. Tout d'abord, nous appelons *support* d'un ensemble de points dans \mathbb{R}^n un hyperplan euclidien tel que tous les points de l'ensemble sont du même côté de l'hyperplan. On note aussi L_∞ la distance infinie ($L_\infty(p, q) = \max_i |p_i - q_i|$). Ainsi :

THEORÈME 5.5 ([Kim84]) *Soit $S \subset \mathbb{Z}^3$. S est un morceau de plan discret si et seulement s'il existe un plan support H de S tel que la L_∞ -distance entre tout point de S et H soit inférieure à 1.*

Énoncé pour la dimension 3, ce théorème est généralisable en dimension supérieure car lié à la définition même des hyperplans discrets. D'un point de vue algorithmique, il est clair que la recherche d'un tel plan support peut être effectuée en s'intéressant à l'enveloppe convexe $CH(S)$ de S . Notons que dans [Kim84, KS91] les auteurs proposent un algorithme de recherche de H en temps quadratique (voir tableau 5.4) mais ce dernier n'est pas correct dans certains cas [Deb95]. Toujours dans ce contexte, nous avons le résultat suivant :

THEORÈME 5.6 ([ST91]) *Soit $S \subset \mathbb{Z}^n$. S est un morceau d'hyperplan discret si et seulement s'il existe un hyperplan euclidien H séparant S et S' , S' étant obtenu par translation de S de 1 selon l'axe principal de H .*

Là encore, le test de séparation de S et de S' peut être résolu en regardant si $CH(S) \cap CH(S')$ est vide ou non. Notons que [ST91] présente un test de séparabilité basé sur la construction d'un problème de programmation linéaire en dimension 3. Pour terminer avec ces caractérisations géométriques, nous considérons l'épaisseur géométrique de S comme étant l'intersection entre l'axe principal de S et l'enveloppe convexe de l'ensemble corde associé à S [GDRZ05]. L'ensemble corde est donné par $\{P - P', \forall P, P' \in S\}$. Ainsi nous avons :

THEORÈME 5.7 ([GDRZ05]) *Soit $S \subset \mathbb{Z}^3$. S est un morceau de plan discret si et seulement si son épaisseur géométrique est inférieure à 1.*

Pour les trois approches précédentes, la combinatoire de l'enveloppe convexe d'un ensemble de points discrets est un élément crucial dans l'analyse de complexité ou tout du moins dans l'efficacité pratique des algorithmes. La figure 5.5 illustre ces différentes approches géométriques.

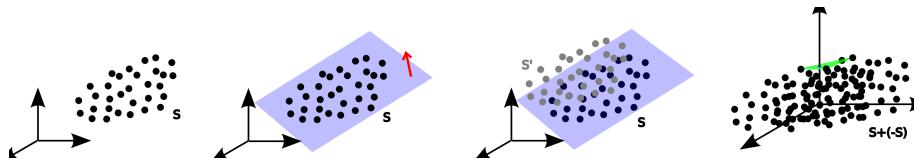


FIGURE 5.5 – Illustration des approches géométriques pour la reconnaissance d'hyperplans discrets : l'ensemble S initial, calcul d'un plan support, test de séparabilité S/S' et calcul de l'épaisseur géométrique de l'ensemble corde de S .

A côté de ces approches, nous avons aussi toutes les techniques issues de la programmation linéaire. En effet, avec la définition 5.3 (resp. la définition 5.4), nous pouvons associer à chaque point discret P de S deux contraintes avec $n+1$ inconnues $\{\gamma_0, \dots, \gamma_n\}$ dans \mathbb{R}^{n+1} (resp. $\{a_1, \dots, a_n, b\}$ dans \mathbb{Z}^{n+1}). Nous avons ainsi deux problèmes LP et LP entière pour décider si S est un morceau d'hyperplan discret ou non. Si nous considérons une orientation spécifique pour l'hyperplan recherché, c'est-à-dire si nous fixons l'axe principal, nous

pouvons nous ramener à un problème LP en dimension n en considérant les inconnues : $\{\gamma_0/\gamma_n, \dots, \gamma_{n-1}/\gamma_n, 1\}$ ou $\{a_1/a_n, \dots, a_{n-1}/a_n, 1, b/a_n\}$ car alors $\gamma_n \neq 0$ et $a_n \neq 0$. Le problème de LP entière devient maintenant un problème LP sur \mathbb{Q} . Nous pouvons donc associer à chaque point discret $P \in S$ dans un espace des paramètres $\{\Gamma_0, \dots, \Gamma_{n-1}\} \in \mathbb{R}^{n-1}$, les contraintes :

$$\Gamma_0 - \frac{1}{2} + P_n + \sum_{i=1}^{n-1} \Gamma_i P_i \leq 0 \quad \text{et} \quad \Gamma_0 + \frac{1}{2} + P_n + \sum_{i=1}^{n-1} \Gamma_i P_i > 0. \quad (5.8)$$

Nous pouvons alors reprendre les résultats de la section 5.3.1 pour résoudre ce problème et ainsi décider si S est un morceau d'hyperplan discret. Par exemple, le résultat de MEGIDDO nous dit que si la dimension est fixe, la reconnaissance d'hyperplan discret est calculable en temps linéaire par rapport au nombre de points discrets.

Dans des dimensions faibles, nous pouvons nous intéresser au calcul de la préimage complète. Pour les dimensions supérieures, la combinatoire de la préimage (polytope de \mathbb{R}^n) ne permet pas d'avoir des algorithmes de mise à jour de celle-ci qui soient efficaces [BY95].

Pour les plans discrets, la préimage est un objet très utilisée [Vit99, VC00, Vee93, Coe02, Siv04, Dex06]. L'un des principaux intérêts étant que la préimage décrit l'infinité des plans euclidiens associés à un morceau de plan discret. Or cette description est primordiale lorsque l'on s'intéresse au problème de la reconstruction réversible (voir chapitre 6). Dans la suite, on note $\mathcal{P}(S)$ la préimage associée à l'ensemble de points discrets S .

L'algorithme est très simple, nous faisons l'hypothèse que l'axe principal du plan est l'axe x_3 (axe Oz). Ensuite, sans perte de généralité, nous initialisons la préimage dans l'espace des paramètres $(\Gamma_0, \Gamma_1, \Gamma_2)$ (notation de l'équation (5.8)) par le cube $[0, 1]^3$. Pour chaque point discret $P \in S$, nous allons mettre à jour la préimage avec les deux contraintes associées (Equ. (5.8)). Plus précisément, nous avons deux calculs d'intersection entre un polyèdre convexe, et une contrainte linéaire. Pour cette mise à jour, nous pouvons utiliser un algorithme linéaire en la taille de la préimage dans le pire cas. Si au cours de l'algorithme $\mathcal{P}(S)$ devient vide, cela signifie qu'il n'existe aucun plan euclidien dont la discréétisation contienne S . Nous concluons alors que S n'est pas un morceau de plan discret.

Cet algorithme incrémental a donc un coût en $O(m \cdot E)$ où $m = |S|$ et E est le nombre d'arêtes de la préimage. Un point crucial consiste donc à donner une borne sur la quantité E . Nous aborderons ce point dans la section suivante. S'il est difficile de parler d'algorithme *output sensitive* pour un algorithme incrémental, à chaque étape, nous avons un processus qui est linéaire en la taille de la sortie de l'étape précédente.

Le tableau 5.4 reprend les différentes approches. Pour plus de précisions sur des algorithmes non détaillés ci-dessus, se référer à [CB06].

Même si l'algorithme de GÉRARD *et al.* [GDRZ05] possède une borne au pire cas très importante ($O(m^7)$), ce dernier est linéaire en pratique. Ce point renforce l'idée que la structure particulière des morceaux de plans discrets joue un rôle crucial dans l'efficacité des algorithmes. C'est cette spécificité discrète que nous analysons dans la section suivante.

5.4.3 Préimage et enveloppe convexe d'un hyperplan discret

Nous cherchons ici à évaluer la combinatoire associée à l'enveloppe convexe de morceaux d'hyperplans discrets et ce pour deux raisons fondamentales :

- l'enveloppe convexe intervient à de nombreuses reprises dans les algorithmes de reconnaissance basés sur la géométrie ;
- dans [Coe02, CSD⁺05], nous avons montré qu'en dimension 3, $|\mathcal{P}(S)| \leq |CH(S)|$.

Commençons par évaluer ces quantités en utilisant des générateurs aléatoires de plans et hyperplans discrets. Pour évaluer $CH(S)$, nous utilisons les outils de la librairie `qhull`¹.

1. <http://www.qhull.org/>

Description	Ref.	Dim.	Complexité	Incrémental
Caclul plan support avec distance < 1	[Kim84, KS91]	3	$O(m^2)$	-
Séparabilité basée sur l'enveloppe convexe	[ST91]	n	$O(m \cdot \log m + m^{\lfloor n/2 \rfloor} + 2^n \cdot m^{2^{n-2}n} \cdot \log m)$	-
Épaisseur de S	[GDRZ05]	3	$O(m^7)$	oui
Algorithme FOURIER-MOTZKIN	[FST96]	3	n.a.	-
LP	[Meg84a]	n fixe	$O(m)$	-
Séparabilité basée sur LP	[ST91]	n fixe	$O(m)$	-
LP entière	[BD05]	n fixe	$O(m \cdot \log D)$	-
Préimage arithmétique	[VC00]	3	$O(mE^2 \cdot \log D)$	oui
Préimage arithmétique	[Coe02]	3	$O(m \cdot \log^2 m)$	oui
Préimage		3	$O(m \cdot E)$	oui
Propriété <i>Evenness</i> (base rectangulaire)	[Vee94]	3	$O(m^2)$	oui
Reconnaissance arithmétique (base rectangulaire)	[Mes02]	3		-

TABLE 5.4 – Solutions algorithmiques pour la reconnaissance de plans et d'hyperplans discrets : m est le nombre de points discrets, n la dimension, E est la taille de la préimage, et D une borne sur la taille du domaine contenant l'ensemble S .

Pour générer un hyperplan discret, nous commençons par générer sa base (ensemble de points discrets dans \mathbb{Z}^{n-1}), nous générions aléatoirement un vecteur normal que nous utilisons pour relever la base et ainsi construire l'hyperplan dans \mathbb{Z}^n . Dans ce qui suit, nous considérons des morceaux connexes contenus dans un domaine borné N^n . Deux générateurs pour la base seront utilisés (voir figure 5.6) :

- base hyper-rectangulaire : on construit un hyper-rectangle de dimension $n - 1$ dont les longueurs des différents côtés sont tirés aléatoirement ;
- base “co-sphérique” : on utilise un générateur de points co-sphérique dans \mathbb{R}^{n-1} fourni avec `qhull`, nous utilisons ensuite `qhull` pour en calculer l'enveloppe convexe et ainsi construire une base discrète comme étant la discréétisation de ce convexe.

Le dernier générateur aura pour objectif de tester des hypothèses pour la classe des morceaux d'hyperplans discrets à base convexe.

Nous avons utilisé ces différents générateurs pour évaluer le comportement de $|CH(S)|$ et $|\mathcal{P}(S)|$ pour cet objet particulier qu'est le plan discret. La figure 5.7 présente les résultats pour le cas de l'enveloppe convexe et pour des dimensions allant jusqu'à 6. On observe que le comportement des courbes semble plus logarithmique que polynomial. On s'aperçoit aussi que bien évidemment, le cas des bases hyper-rectangulaires est plus *simple* que le cas co-sphérique.

Dans la figure 5.8, nous avons évalué la taille de la préimage sur ces objets générés. Pour ce test, le temps de calcul devient vite très important en fonction de la dimension. C'est la raison pour laquelle il y a moins d'échantillons que pour l'enveloppe convexe. Nous pouvons cependant remarquer des comportements logarithmiques et globalement une taille très faible de cette préimage dans le cas des morceaux à base hyper-rectangulaire.

Pour mettre cela en avant et en se restreignant à la dimension 3, la figure 5.9 compare les deux structures ($CH(S)$ et $\mathcal{P}(S)$) pour la dimension 3. Si on peut observer que la combinatoire est globalement faible (< 300 sommets pour près de 80000 points discrets), la taille de la préimage est encore plus réduite (< 60 sommets pour près de 80000 points discrets).

Pour expliquer formellement ces résultats, nous utilisons les outils précédents. Pour cela, considérons l'écriture analytique des hyperplans et notons $\alpha = \max\{|a_1|, \dots, |a_n|\}$. Dans

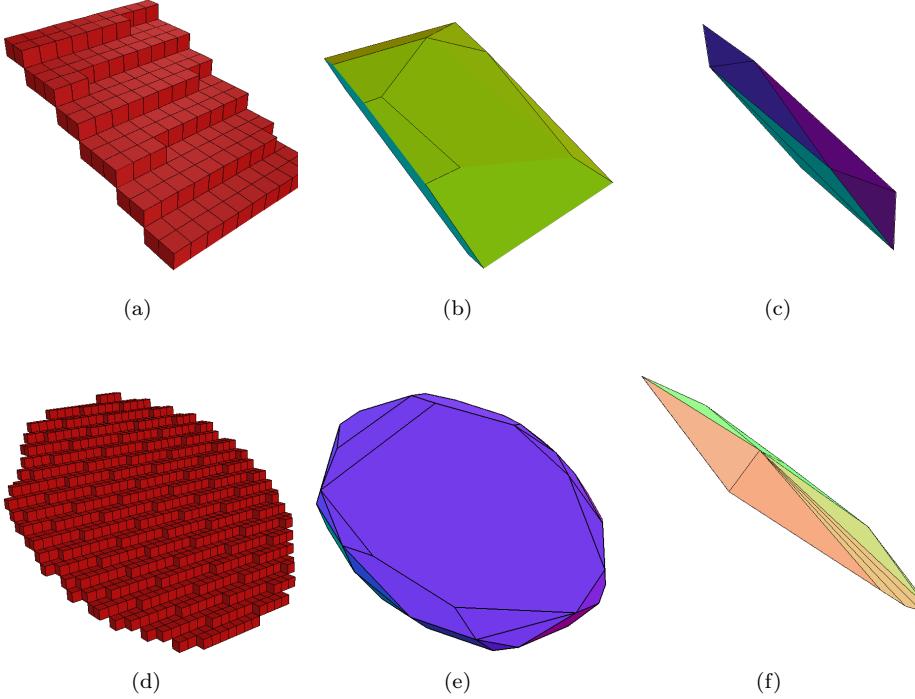


FIGURE 5.6 – Illustration de deux morceaux de plans discrets selon les générateurs proposés : (a) un morceau ayant une base rectangulaire (b) son enveloppe convexe et (c) sa préimage ; (d) un morceau de plan à base “co-sphérique”, (e) son enveloppe convexe et (f) sa préimage.

toutes les techniques de reconnaissance, cette quantité est généralement proportionnelle au diamètre de S que nous pouvons borner par $O(N)$.

THEORÈME 5.8 ([CB06]) 1. Si S est un morceau d’hyperplan discret dont la base est hyper-rectangulaire, alors

$$|CH(S)| \leq c_n \log^{n-1}(1 + \alpha); \quad (5.9)$$

2. Sinon,

$$|CH(S)| \leq c_n N^{\frac{(n-1)^2}{n+1}}; \quad (5.10)$$

3. Si S est un morceau de plan discret ($n = 3$) dont la base est un convexe discret, ou la discréttisation d’un convexe continu dans les hypothèses du théorème 5.4, alors :

$$|CH(S)| \leq cN^{\frac{2}{3}} \log^2(1 + \max\{N, \alpha\}). \quad (5.11)$$

c, c_n sont des quantités dépendant uniquement de la dimension n .

Preuve : Les détails de la preuve sont donnés dans [CB06]. Cependant, nous précisons ici les différentes utilisations des outils précédents pour obtenir ces résultats :

- Point 1 : si un morceau d’hyperplan a une base rectangulaire, nous pouvons coder S comme le résultat d’un problème de LP entière avec $2n+2$ contraintes ($2n$ pour coder la base et 2 pour la “pente” de l’hyperplan). Le résultat découle donc du théorème 5.2 avec $m = 2n + 2$.

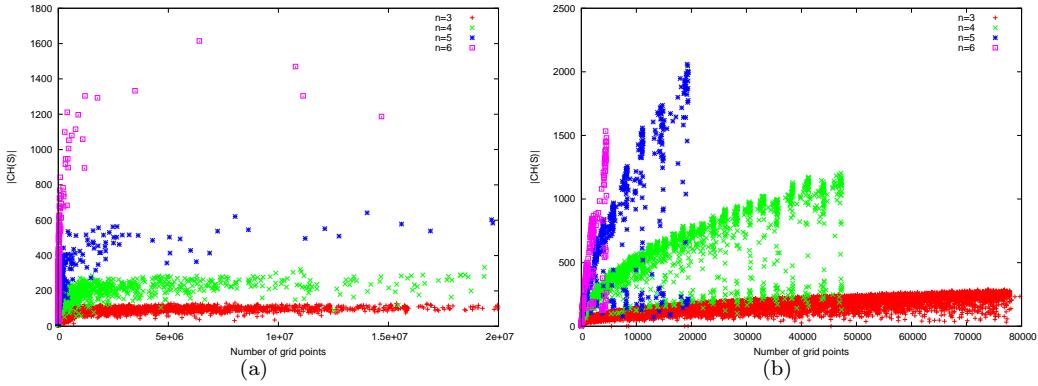


FIGURE 5.7 – Analyse expérimentale du nombre de sommets de $CH(S)$ pour les générateurs à base hyper-rectangulaire (a) et co-sphérique (b) pour $n = \{3, 4, 5, 6\}$.

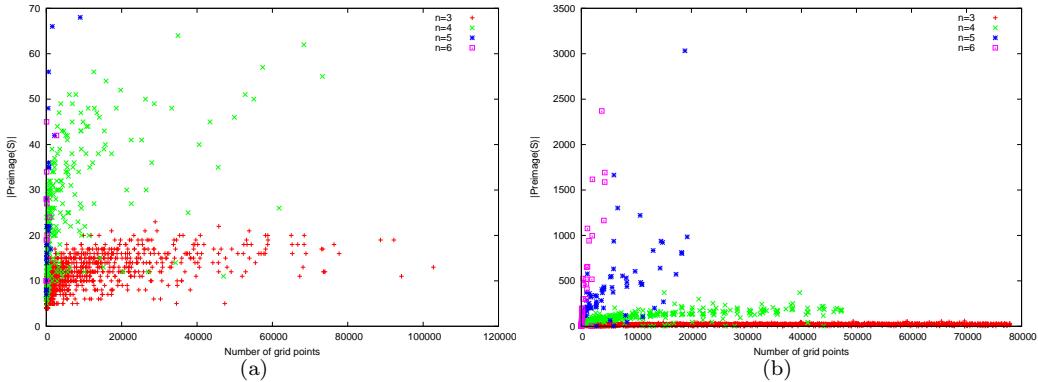


FIGURE 5.8 – Analyse expérimentale du nombre de sommets de $\mathcal{P}(S)$ pour les générateurs à base hyper-rectangulaire (a) et co-sphérique (b) pour $n = \{3, 4, 5, 6\}$.

- Point 2 : nous sommes ici dans le cas général, $(\text{Vol } CH(S))$ est borné par $O(N^{n-1})$ et nous utilisons le théorème 5.3 pour conclure.
- Point 3 : en dimension 3, le théorème 5.4 et plus précisément l'équation 5.6 nous permet de définir S comme un problème de LP entière avec $N^{2/3}$ contraintes pour la base, et deux pour la pente. Le résultat découle de 5.2.

□

En conclusion, de cette partie, nous pouvons faire deux remarques, l'une théorique sur la nature des hyperplans, et l'autre pratique. Si le théorème 5.8 nous donne quelques bornes sur $CH(S)$ et donc sur $\mathcal{P}(S)$ pour $n = 3$, les expériences nous montrent que nous sommes encore loin d'obtenir des bornes fines. Il reste des spécificités discrètes encore mal appréhendées dans les analyses.

En pratique et en dimension 3, l'objet géométrique $\mathcal{P}(S)$ se révèle comme étant très intéressant car même si des algorithmes naïfs d'intersection plan/polyèdre sont utilisés pour la reconnaissance basée sur la préimage, la taille de cet objet est très faible et donc permet la conception d'algorithmes très rapides. Quelque part, cette faible taille de préimage suggère aussi que des techniques plus évoluées d'intersection plan/polyèdre convexe, par exemple en utilisant un représentation hiérarchique du polyèdre [DK83, GO97], risquent, certes d'améliorer la complexité asymptotique, mais sans doute de diminuer l'efficacité pratique de la reconnaissance avec des structures trop lourdes à manipuler.

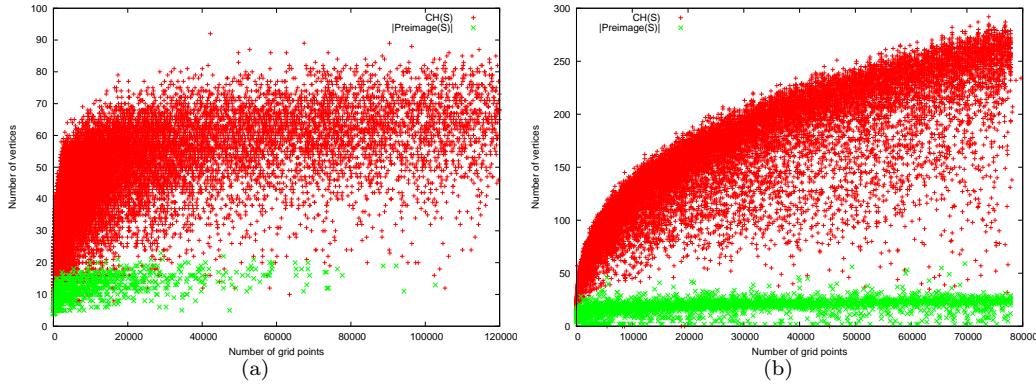


FIGURE 5.9 – Analyse comparée de $|CH(S)|$ et $|\mathcal{P}(S)|$ pour la dimension 3 pour les plans discrets à base hyper-rectangulaire (a) et co-sphérique (b).

Pour les algorithmes utilisant exclusivement l'enveloppe convexe, ces résultats suggèrent de s'orienter vers les techniques *output sensitive*.

5.4.4 Cercles discrets

La reconnaissance de cercle discret a toujours été un problème classique de géométrie discrète. Une des raisons principales est que le problème du cercle est une continuation logique une fois les premiers résultats obtenus sur les droites discrètes (voir [CGRT04] pour une bibliographie complète).

Globalement, peu d'algorithmes ont su tirer parti de la spécificité des objets discrets ou de propriétés arithmétiques des cercles (l'ordre 2 rendant les choses beaucoup plus complexes [HW75]). Ainsi, certaines techniques construisent un problème de séparabilité classique en tant que problème de LP classique rendant le calcul linéaire avec [Meg84b].

Dans cette section, nous présentons quelques résultats permettant d'intégrer un peu de spécificité discrète dans les complexités des algorithmes de séparabilité qui seront *in fine* utilisés pour la reconnaissance.

Rappelons le problème : considérons un ensemble discret S . Pour décider si S est un disque discret, nous allons tester la séparabilité de S et de son complémentaire $\bar{S} = T$ par un cercle.

De manière triviale, nous pouvons réduire S à son enveloppe convexe $CH(S)$ et cela ne change rien concernant la séparabilité (voir figure 5.10). Ensuite, nous pouvons énoncer un filtrage des points de T en utilisant le principe illustré dans la figure 5.11 : supposons le triangle formé par $s, s' \in CH(S)$ et $t \in T$, s'il existe $t' \in T$ à l'intérieur de ce dernier, nous pouvons supprimer le point t de T sans que cela ne change la séparabilité. Pour effectuer cette recherche de triangle ne contenant aucun point de T , nous procéderons en analysant indépendamment toutes les arêtes de $CH(S)$ [CGRT04].

La spécificité discrète intervient au niveau du filtrage de T , avec en préalable la définition suivante (voir figure 5.12) :

DÉFINITION 5.6 (Points de BEZOUT) Soit un segment discret $[mn]$ défini par un vecteur directeur \vec{u} sous forme réduite ($\text{pgcd}(u_x, u_y) = 1$), un point P est un point de BEZOUT de $[mn]$ si :

$$\overrightarrow{mP} = \vec{v} + k\vec{u} \quad \text{avec } k \in \mathbb{Z} \quad \text{et} \quad \det(\vec{u}, \vec{v}) = \pm 1$$

De plus, P est le point le plus proche de la médiatrice de $[mn]$.

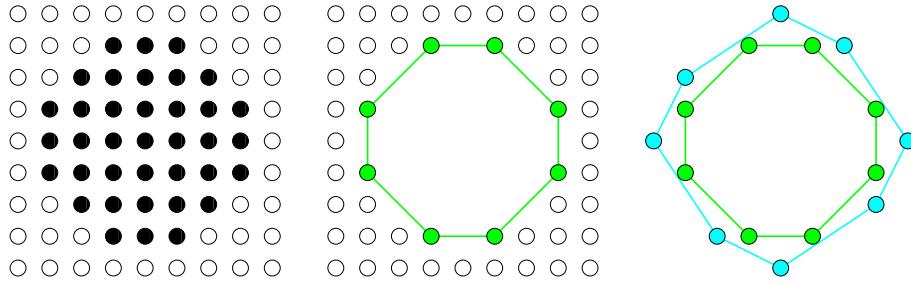


FIGURE 5.10 – Illustration du processus global de filtrage pour la reconnaissance de disques discrets.

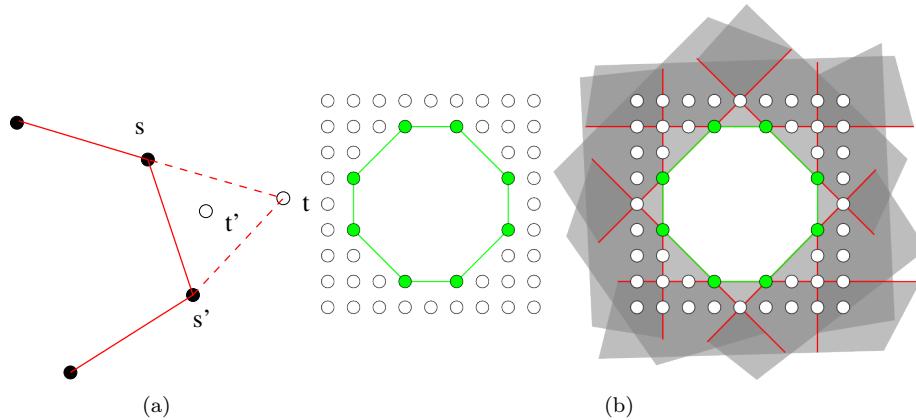


FIGURE 5.11 – Illustration du filtrage de T : test d'inclusion dans un triangle (a) et décomposition du test, arête par arête (b).

Par cette définition, un segment $[mn]$ possède un point de BEZOUT de part et d'autre du segment. Si nous orientons le segment $[mn]$, on considère par définition qu'un seul de ces deux points, celui appartenant au demi-espace défini par $[mn]$ pour une certaine convention d'orientation. D'un point de vue algorithmique, le vecteur \vec{v} peut être donné par l'algorithme d'EUCLIDE étendu [HW75]. Le coût de cet algorithme est proportionnel à $\log(\min(|u_x|, |u_y|))$. L'existence d'un tel vecteur étant donné par l'identité de BEZOUT classique. Basé sur cette définition, nous avons le résultat :

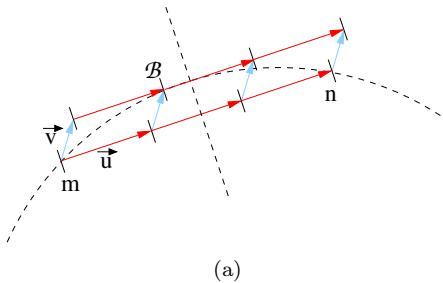
LEMME 5.2 ([CGRT04]) *Considérons un segment discret orienté $e = [mn]$ avec $m, n \in \mathbb{Z}$ (par exemple m à gauche de n en abscisse) et notons T_e l'ensemble des points discrets de T appartenant au demi-espace associé à e (par exemple le demi-espace au dessus de $[mn]$). Les seuls triangles (m, n, t) avec $t \in T_e$ ne contenant aucun autre point de T_e sont ceux engendrés par les points de BEZOUT.*

Pour conclure sur ce filtrage, nous avons le lemme suivant décrivant complètement la figure 5.10 :

LEMME 5.3 ([CGRT04]) *Soit S un objet discret, S est un disque discret si et seulement si S est séparable de T par un cercle sachant :*

- $S = CH(S)$
- T est l'ensemble des points de BEZOUT (un par arête de $CH(S)$).

De plus, en utilisant le théorème 5.4 ou plus simplement l'équation (5.6) et en considérant



(a)

FIGURE 5.12 – Illustration de la définition des points de BEZOUT associés à un segment discret.

que S est dans une fenêtre $N \times N$, nous obtenons :

$$|S| = O(N^{2/3})$$

$$|T| = O(N^{2/3})$$

Notons que des bornes similaires existent en remplaçant N par le diamètre de S .

Maintenant que l'arithmétique nous a permis de réduire les ensembles S et T , nous pouvons reprendre les coûts des algorithmes classiques de séparabilité pour en énoncer des variantes discrètes. Le tableau 5.5 détaille l'algorithme optimal se basant sur le calcul d'enveloppe convexe discret de VOSS. Rappelons que h désigne la taille du convexe résultant de l'algorithme. Ainsi, grâce aux résultats sur les polytopes discrets, $h = O(N^{2/3})$. Il est intéressant de noter que les algorithmes basés sur de la programmation linéaire en dimension 3 sont linéaires.

Pour le problème de reconnaissance de cercles, la préimage peut se définir comme étant le lieu des centres des cercles séparants (les rayons peuvent être déduits par la suite). Avec la formulation comme un problème linéaire de dimension 3 de [OKM86], ce lieu peut être défini comme le domaine de satisfaction du problème linéaire. En dimension 2, ce lieu est donné par l'intersection de demi-espaces (définis par les médiatrices de $[st]$ pour tout couple $s \in S$ et $t \in T$), cette préimage est une région convexe potentiellement non bornée [CGRT04].

Enveloppe convexe [Vos93] Pour chaque arête, calcul du point de BEZOUT Décision LP dimension 3 ou Préimage LP dimension 3 ou Préimage LP dimension 2	$O(N)$ $O(h \log \delta(X))$ $O(h)$ $O(h \log h)$ $O(h^2 \log h)$
Bilan avec décision LP dimension 3 Bilan avec préimage LP dimension 3 Bilan avec préimage LP dimension 2	$O(N + N^{2/3} \log(N))$ $O(N + N^{2/3} \log(N))$ $O(N^{4/3} \log(N))$

TABLE 5.5 – Algorithme générique pour la reconnaissance de disques discrets et analyses de complexités pour les différentes options.

Au delà de ces résultats théoriques, en pratique nous avons surtout des problèmes de reconnaissance de cercle sur des portions de contour de S . Dans ce cas, l'algorithme global d'enveloppe convexe pourrait ne pas être utilisable si S possède des points d'inflexion. Un autre objectif serait d'avoir une notion de couverture tangentielle pour les arcs de cercles. En d'autres termes, le problème consiste, en tout point du contour discret, à reconnaître l'arc de cercle discret maximal (non extensible à gauche et à droite). Cette structure est très utilisée pour les segments discrets [FT99, dV07].

Pour cela, il nous faut utiliser un algorithme d'enveloppe convexe dynamique [Mel87, Buz07], ensuite pour chaque arête nouvellement créée, il faudra utiliser un algorithme

d'EUCLIDE pour calculer le point de BEZOUT et ainsi mettre à jour S et T . Pour terminer, il nous faut un solveur de LP qui soit lui aussi dynamique, c'est-à-dire avec ajout et suppression de contraintes.

Pour l'instant, nous nous sommes intéressés dans [CGRT04] à un algorithme incrémental (seulement ajout de contrainte) basé sur la mise à jour de la préimage en LP_2 . Si la complexité est supérieure que les techniques LP_3 , certains points sont incrémentaux et plus simples à implémenter.

5.5 Conclusion

Dans ce chapitre, nous avons abordé les différents problèmes de reconnaissance de formes discrètes élémentaires que sont les droites, les cercles et les plans/hyperplans discrets.

Parmi les solutions algorithmiques présentées, nous avons pu nous rendre compte de l'importance des résultats d'autres domaines comme la géométrie algorithmique, l'arithmétique, la programmation linéaire, Cette constatation justifie pleinement notre cadre de travail qui consiste à nous placer à l'interface de plusieurs disciplines afin de proposer des algorithmes efficaces. Pour synthétiser un peu ces travaux, nous pouvons constater que la droite discrète est maintenant un objet pour lequel les connaissances arithmétiques sont suffisamment abouties. Le challenge n'est plus réellement lié à la droite discrète analytique mais plutôt à des extensions comme par exemple les droites épaisses, bruitées ou fondées sur une autre formulation.

Pour les plans et hyperplans discrets, nous avons présenté des analyses permettant de mieux comprendre et de mieux intégrer les spécificités discrètes dans les algorithmes de reconnaissance utilisant *in fine* des outils de géométrie algorithmique. A la vue des différentes expérimentations, nous pouvons remarquer que d'autres résultats sont à espérer.

Le cercle discret est sans doute un exemple très représentatif du gain que l'on a à considérer les propriétés arithmétiques induites par la structure discrète dans les outils de séparabilité pour reconnaître des disques ou cercles discrets.

Que ce soit pour les cercles discrets ou pour les plans, si nous maîtrisons un peu mieux les coûts théoriques des algorithmes, l'implémentation et la mise en œuvre de ces techniques est un problème non négligeable. Plus précisément, le résultat de MEGIDDO, que nous utilisons souvent comme un argument *massue* pour conclure sur la complexité, n'est pas implantable à partir de la dimension 3. Pour conclure, des efforts sont encore à mener pour rapprocher la complexité asymptotique de l'efficacité en temps des algorithmes, que ce soit d'un point de vue théorique ou pratique.

Chapitre 6

Reconstruction réversible d'objets discrets par des parties linéaires

Sommaire

6.1	Introduction	66
6.2	Reconstruction 2D sur grille isothétique irrégulière	66
6.2.1	Contexte et éléments préliminaires	66
6.2.2	Reconstruction géométrique et topologique couplée	67
6.2.3	Raffinement et mise à jour locaux	69
6.3	Reconstruction réversible en dimension 3	73
6.3.1	Éléments préliminaires	73
6.3.1.1	Algorithme des <i>Marching-Cubes</i>	73
6.3.1.2	Segmentation d'une surface discrète en plans discrets maximaux	74
6.3.2	Simplification et Optimisation	75
6.3.2.1	Triangles homogènes	75
6.3.2.2	Triangles non-homogènes	75
6.3.3	Algorithme global de simplification	76
6.3.4	Résultats et discussion	77
6.4	Conclusion	77

6.1 Introduction

Le chapitre précédent s'intéressait au problème du passage d'une description en extension vers une description en compréhension des objets élémentaires. Dans ce chapitre, nous passons à l'étape suivante qui consiste à la modélisation ou reconstruction réversible d'une forme euclidienne dans le sens où la discréétisation nous donne exactement l'ensemble initial. Comme nous l'avons précisé dans le chapitre 3, les objets fondamentaux du chapitre précédent seront grandement réutilisés.

Nous présentons tout d'abord un schéma générique de reconstruction réversible en dimension 2. Ce schéma est générique puisqu'il est défini sur les grilles isothétiques irrégulières, pouvant s'instancier en de nombreuses grilles (grilles classiques, de subdivision, ...). Le contexte applicatif de cette approche est la reconstruction sur des structures discrètes issues d'un solveur en arithmétique d'intervalles.

Ensuite, nous abordons la reconstruction réversible en dimension 3 en présentant une technique *bottom-up* basée sur la simplification d'une triangulation issue des *Marching-Cubes* en intégrant des informations géométriques données par la segmentation en plans discrets de la surface.

6.2 Reconstruction 2D sur grille isothétique irrégulière

6.2.1 Contexte et éléments préliminaires

Dans ce paragraphe, nous allons revenir sur le solveur d'intervalles présenté dans l'algorithme 1 sur l'arithmétique $\mathbb{I}_{\mathbb{F}}$ en dimension 2 (voir chapitre 4). Le problème consiste à trouver un encadrement des solutions d'une fonction implicite $f(x, y) = 0$. La figure 6.1 illustre la décomposition récursive d'un domaine initial, les cellules grises correspondant aux intervalles de dimension 2 de la forme $X \times Y$ tels que $0 \in \square f(X, Y)$ selon plusieurs valeurs du paramètre ϵ pour le critère d'arrêt (voir section 4.3.2). On se rend compte tout d'abord de la subdivision hiérarchique du domaine, proche d'une analyse en *quadtree* [Sam90].

Le problème que nous cherchons à résoudre est d'obtenir une représentation polygonale valide au sens de l'arithmétique d'une fonction pour laquelle nous faisons l'hypothèse que la solution est de dimension 1. La validité par rapport à l'arithmétique consiste à trouver une représentation contenue dans l'ensemble des cellules solutions. Dans [Sny92], une première solution est proposée mais est peu satisfaisante par rapport à la taille de la reconstruction polygonale créée (une arête créée par cellule solution) et par rapport aux hypothèses sur la topologie de l'objet continu sous-jacent restreignant ainsi les cas où l'algorithme peut être utilisé.

Ce que nous observons dans la figure 6.1, c'est que la topologie de l'ensemble des cellules peut être complexe et varie en fonction de la résolution choisie. D'une manière générale et sans plus d'hypothèses, nous ne pouvons pas lier la topologie discrète des cellules avec la topologie de l'objet continu sous-jacent. Pour établir un tel lien en posant des hypothèses sur la forme sous-jacente, nous pouvons exploiter les propriétés de convergence des solveurs [Sny92], ou encore des outils plus proches de la géométrie discrète comme la *par(r)*-régularité [LCG98] ou le *r*-homéomorphisme [SK05].

De manière pragmatique, il nous faut un outil de polygonalisation de courbes discrètes irrégulières (voir section 3.3.2) sur des objets complexes. Dans un premier temps, nous avons présenté dans [CZ06] un algorithme de reconstruction d'arc polygonal contenu dans une (1)-courbe irrégulière. Cet algorithme incrémental linéaire ($O(1)$ par incrément) se base sur un calcul très simple de visibilité interne à la (1)-courbe (figure 6.2). Notons que des algorithmes moins restrictifs sur la classe des *droites discrètes* considérées peuvent aussi être utilisés, mais la complexité ne serait plus linéaire [Coe05, Dex06].

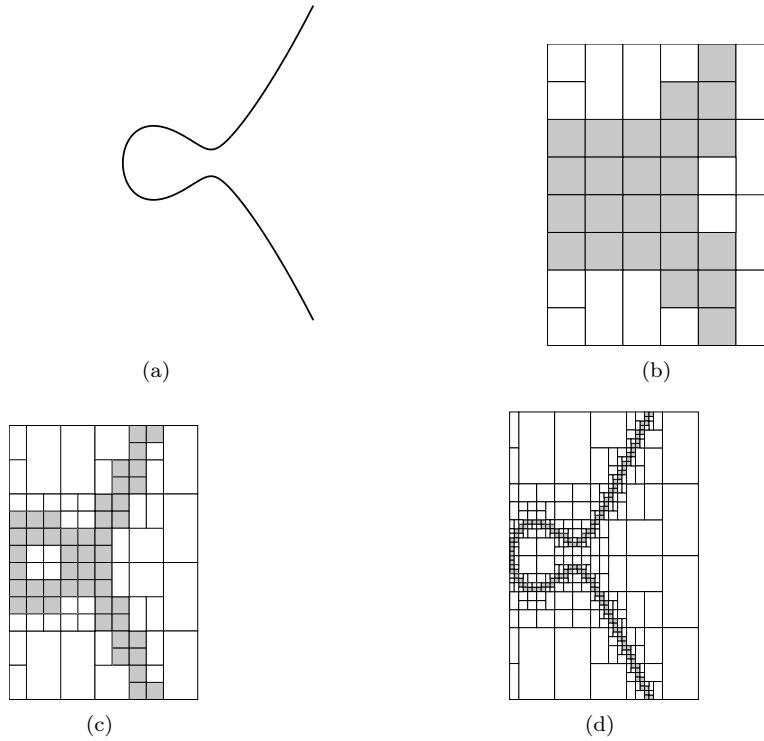


FIGURE 6.1 – Évaluation de la fonction $f(x, y) = y^2 - x^3 + x - \frac{1}{2}$ sur l'intervalle $[-4, 4] \times [-4, 4]$ avec comme critère d'arrêt, une borne sur la largeur des intervalles fixée à : (b) 1.0, (c) 0.5 et (d) 0.1.

6.2.2 Reconstruction géométrique et topologique couplée

Notre approche se base sur une généralisation du modèle de supercouverture au grilles discrètes irrégulières vérifiant les propriétés (4.1), (4.2), (4.3) et (4.4) des modèles analytiques, mais perdant dans sa définition l'approche par morphologie mathématique qui ici n'a pas de sens [CZ06]. Pour traiter les objets topologiquement complexes et « épais », il

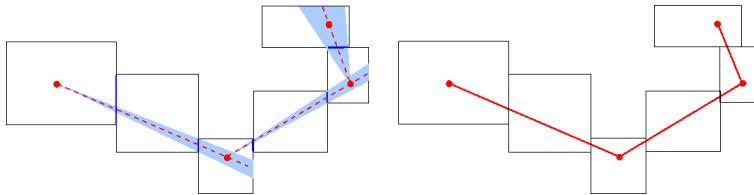


FIGURE 6.2 – Illustration de la reconstruction par une courbe polygonale d'une (1)-courbe par la propagation d'un cône de visibilité.

nous faut un processus pour extraire des portions de courbes sur lesquelles nous utiliserons l'algorithme précédent. Pour cela, nous allons procéder comme suit :

1. on choisit une direction h (fonction de hauteur dans la théorie de REEB, voir figure 6.3-a), par exemple, parallèle à un axe;
2. on construit un graphe de REEB en caractérisant les points critiques suivants, lors du parcours selon la direction entre les instants t et $t + 1$:
 - *begin (b)* : début d'une composante connexe en $t + 1$;
 - *end (e)* : fin d'une composante connexe en $t + 1$;

- *merge (m)* : fusion à $t + 1$ de deux composantes (ou plus) de t ;
 - *split (s)* : décomposition d'une composante à t en plusieurs composantes à $t + 1$;
3. on utilise les arêtes du graphe précédent pour définir les différentes (1)-courbes.

Supposons l'ensemble de cellules connexes donné dans la figure 6.3-(b). Pour transformer cet objet en (1)-courbe, nous utilisons un recodage des cellules guidé par la direction h choisie (figure 6.3-b). Le recodage change la configuration des cellules (en fonction de h) mais ne change pas le domaine total de l'objet.

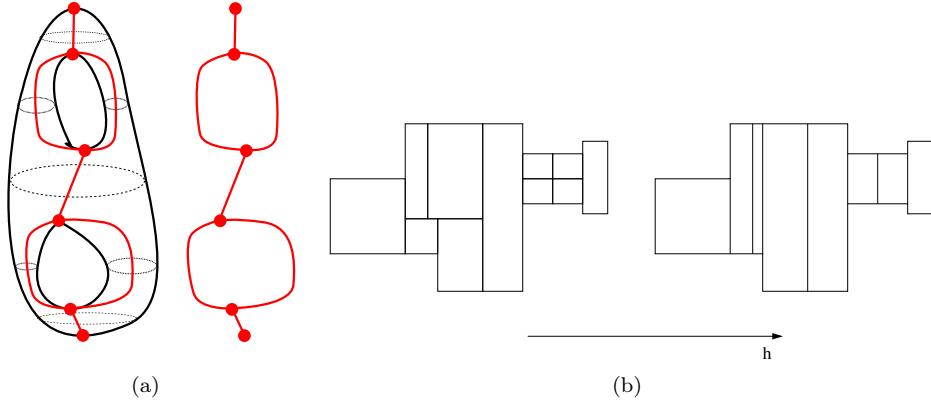


FIGURE 6.3 – Représentation classique du graphe de REEB en géométrie différentielle basée sur les points critiques de MORSE (a). Illustration du recodage d'un (1)-objet en (1)-courbe en fonction d'une fonction de hauteur h .

En pratique, la construction du graphe de REEB et le recodage de ses arêtes se font dans le même temps : si h correspond aux x croissants, les cellules sont triées selon leur bord gauche et sont considérées une par une dans cet ordre. A chaque étape, le graphe est mis à jour et le recodage se fait à la volée. La figure 6.4 illustre cela sur un petit exemple, pour plus de détails, l'algorithme complet est détaillé dans [VCT06b].

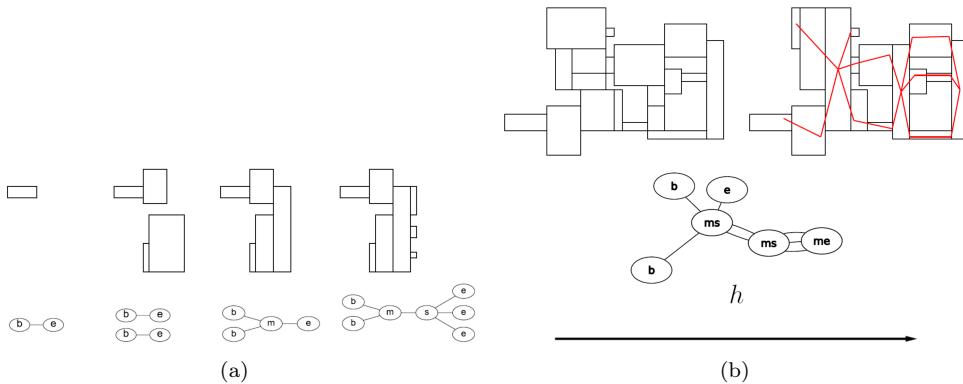


FIGURE 6.4 – (a) Exemple de construction parallèle du graphe de REEB et du recodage des cellules pour former des (1)-courbes pour les arcs du graphe. (b) Illustration complète avec reconstruction polygonale.

Le graphe, mis à part les invariants topologiques que nous pouvons obtenir, nous permet donc de faire une segmentation en (1)-courbes de notre objet (rappelons qu'il est supposé représenter un objet continu de dimension 1). Pour obtenir une reconstruction centrée et réversible, nous faisons le choix de fixer un point de départ au centre de toutes les cellules de type (*s*) et (*m*). Ensuite, les reconstructions sont lancées en parallèle sur les arcs adjacents à ces cellules, vers les extrémités ou d'autres nœuds (*s*) et (*m*). Enfin, lorsque deux

reconstructions se *rejoignent* (cas par exemple où les extrémités d'une (1)-courbe ont pour types (*s*) et (*m*)), nous ajoutons arbitrairement un petit segment pour relier les deux courbes polygonales en cours de calcul (voir figure 6.5). Remarquons qu'avec ce schéma, si l'objet discret possède des symétries, la reconstruction sera bien centrée et symétrique.

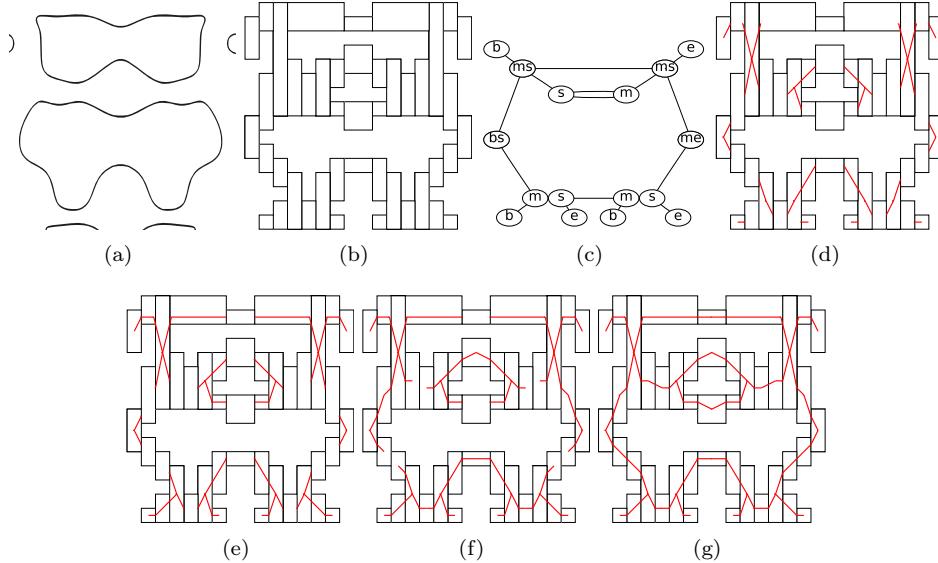


FIGURE 6.5 – Tracés de la fonction $x^2 + y^2 + \cos(2\pi x) + \sin(2\pi y) + \sin(2\pi x^2) \cos(2\pi y^2) = 1$ sur $[-1, 1]^2$ (exemple de [Sny92]) : (a) approximation du tracé, (b) cellules résultantes de l'approximation en intervalles et du recodage, (c) le graphe de REEB associé. Les figures (d) à (g) illustrent les étapes de la reconstruction polygonale partant des nœuds du graphe de REEB.

Au final, nous avons une reconstruction :

- **valide par rapport à l'arithmétique d'intervalles** : le modèle de supercouverture, intrinsèque à la reconstruction des (1)-courbes, contraint la reconstruction à être intérieure au domaine solution du solveur ;
- **valide par rapport à la topologie de l'objet discret** : l'utilisation du graphe de REEB permet une reconstruction homéomorphe à l'objet discret irrégulier.

Les figures 6.6 et 6.7 présentent quelques résultats de reconstruction approximant des courbes implicites par l'usage de l'arithmétique d'intervalles.

6.2.3 Raffinement et mise à jour locaux

Par la suite, [VCT06a] présente une extension à ce travail permettant des modifications locales sur la structure de REEB et la reconstruction. Plus précisément, nous considérons deux opérateurs sur les cellules :

- **raffinement local d'un ensemble de cellules** : par exemple, ce processus peut être guidé par une itération supplémentaire dans la résolution recursive de l'algorithme 1. L'hypothèse que nous formulons sur ce processus est une propriété d'inclusion : le résultat du raffinement doit être inclus dans l'ensemble de cellules raffiné ;
- **fusion** : regroupement dans une même cellule d'un ensemble de cellules. Là encore, nous utilisons des hypothèses liées à l'arithmétique d'intervalles : le résultat du regroupement consistera en une cellule correspondant à la boîte englobante de l'ensemble de cellules initial.

En résultat de [VCT06a], chacun de ces opérateurs modifiera tout d'abord localement le graphe de REEB (uniquement au niveau des arcs impliqués dans la modification), puis

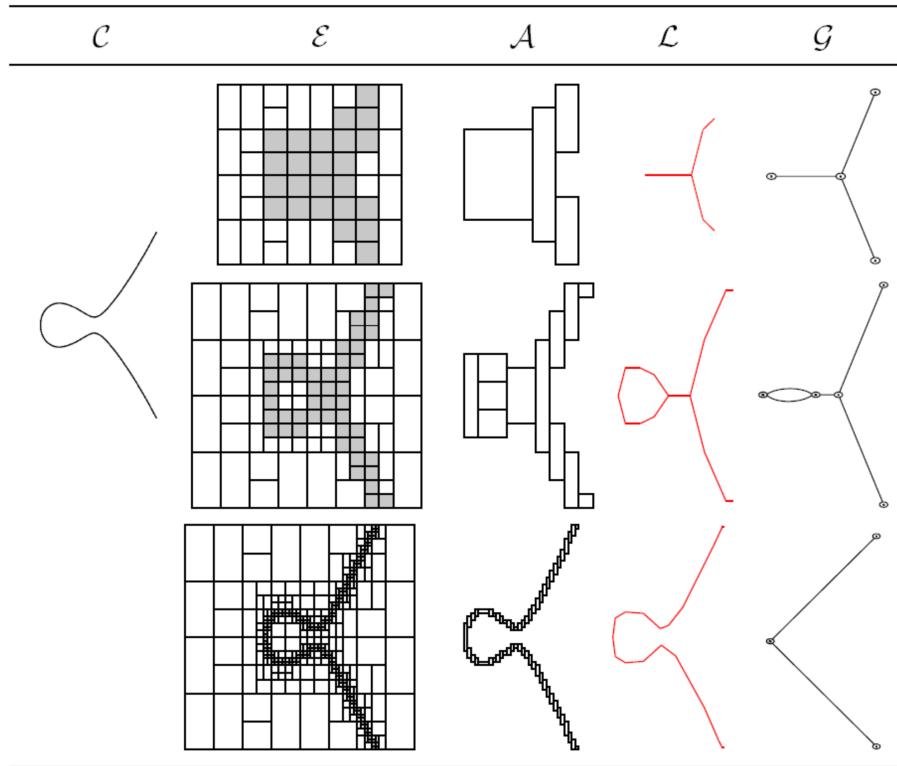


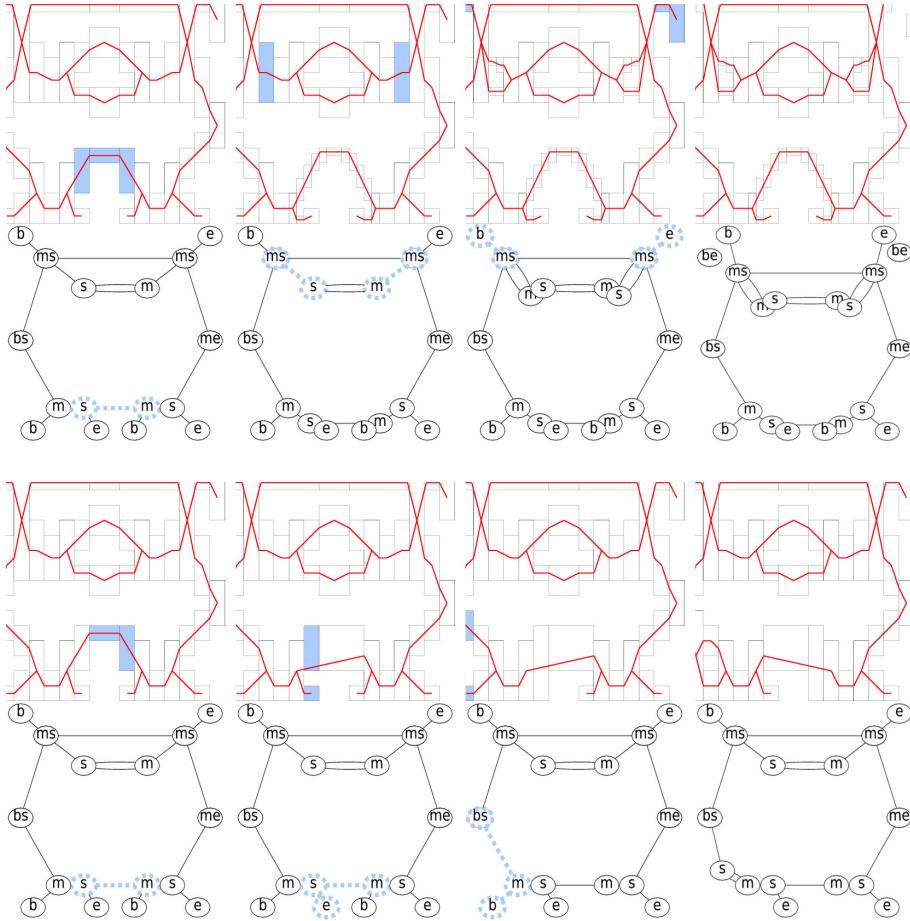
FIGURE 6.6 – Illustration complète pour la fonction $f(x, y) = y^2 - x^3 + x - \frac{1}{2}$ sur l'intervalle $[-4, 4] \times [-4, 4]$ (\mathcal{C} , voir figure 6.1). \mathcal{L} correspond aux reconstructions polygonales et \mathcal{G} , au graphe de REEB associé.

un phase de reconstruction locale aux (1)-courbes impliquées dans les modifications (voir figure 6.7).

En pratique, la nature des applications de ces opérateurs est très générique : dans un contexte de modélisation, une sélection assistée par ordinateur interactif peut être envisagée. Dans un système automatique, nous pouvons appliquer ces opérateurs pour une rétroaction sur le calcul en cours : on utiliserait l'algorithme 1 et la reconstruction polygonale [VCT06b] pour obtenir une première solution. Ensuite, nous pouvons analyser la géométrie (ou même la topologie) de la reconstruction pour décider, s'il y a lieu, de raffiner certaines zones du calcul ou simplifier certaines parties. L'intérêt de reboucler sur le processus de reconstruction précédent est que nous maintenons ainsi les propriétés d'exactitude par rapport à l'arithmétique, en incluant un contrôle topologique fin. Pour illustrer cela, nous faisons dans [VCT07] une évaluation de la courbure sur la courbe polygonale pour décider d'un raffinement ou non.

En dehors du contexte de l'arithmétique d'intervalles, tous ces outils peuvent être utilisés dans le cas de la reconstruction réversible d'objets discrets définis sur la grille régulière. La figure 6.8 présente par exemple une reconstruction réversible classique dans une courbe (1)-connexe. Dans ce cas régulier, comme dans le cas isothétique irrégulier, d'autres algorithmes de reconnaissance et de reconstruction peuvent être envisagés. C'est le cas par exemple des algorithmes basés sur la notion de préimage généralisée (voir chapitre 5 et [Dex06]). Cependant, la gestion des intersections complexes via la structuration en graphe de REEB offre une solution originale à ce problème.

Pour illustrer cela, ainsi que la fiabilité du processus lors d'un passage à l'échelle, la figure 6.9 présente un exemple de reconstruction complexe.



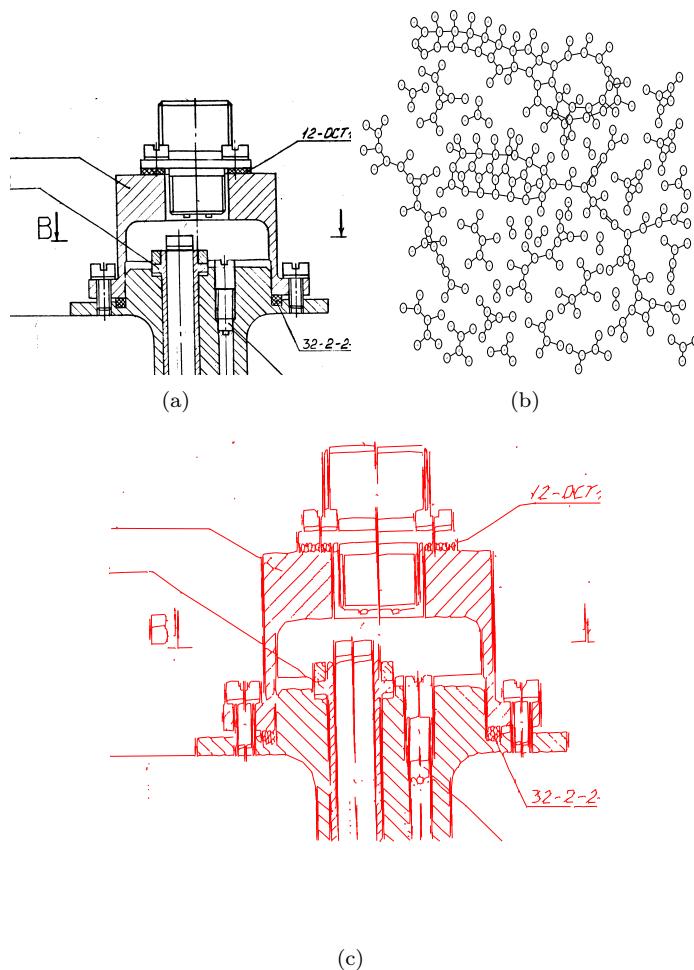


FIGURE 6.9 – Exemple de reconstruction d'un objet issu d'un document numérisé : (a) image initiale, (b) graphe de REEB représentant la forme et (c) la reconstruction.

6.3 Reconstruction réversible en dimension 3

Dans cette section, nous revenons sur le cas discret régulier classique pour présenter une méthode de polyédration réversible en dimension 3. Comme présenté dans la section 3.4.2, nous détaillons une approche *Bottom-Up* basée sur une triangulation initiale donnée par l'algorithme des *Marching-Cubes*.

6.3.1 Éléments préliminaires

6.3.1.1 Algorithme des *Marching-Cubes*

Supposons une image volumique $V(x, y, z)$ à valeurs dans \mathbb{R} pour tout point $(x, y, z) \in \mathbb{Z}^3$. L'algorithme des *Marching-Cubes* (noté MC) a été proposé initialement par LORENSEN et CLINE [LC87] et consiste en la construction d'une isosurface triangulée. Initialement, l'objectif de ces travaux était la visualisation rapide de surfaces en imagerie médicale. Pour définir la surface, nous allons nous intéresser aux cellules de la grille. Une telle cellule de position (x, y, z) est définie comme les 8 points discrets $(x+i, y+j, z+k)$ avec $i, j, k \in \{0, 1\}$. L'isosurface est construite localement sur une cellule et propagée aux cellules voisines en se basant sur des configurations élémentaires. Selon les rotations et symétries possibles des cellules, [LC87] définissent ainsi 14 configurations (voir figure 6.10). Notons cependant que ce jeu de configurations implique des ambiguïtés dans la reconstruction et donc potentiellement des trous sur la surface obtenue. Dans ce qui suit, nous utilisons les résultats de LACHAUD et MONTANVERT permettant de faire le lien entre la topologie de l'isosurface et celle de la surface discrète que l'on considère [Lac96, LM00].

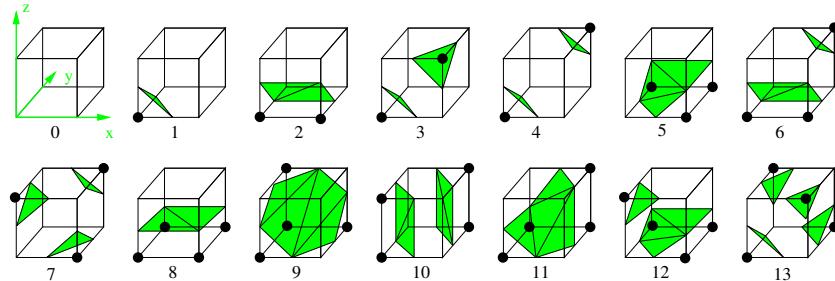


FIGURE 6.10 – Les 14 configurations de l'algorithme initial des *Marching-Cubes*.

Supposons maintenant l'objet discret $V : \mathbb{Z}^3 \rightarrow \{0, 1\}$. Nous supposons l'objet (2)-connexe et nous définissons sa surface comme étant issue des adjacences de JORDAN (18, 6) [CMC07, chap.3]. Dans ce qui suit, nous considérons l'algorithme des MC adapté à cette surface et donc sans ambiguïté [Lac96, LM00]. Dans ce cas, nous avons le résultat suivant :

LEMME 6.1 ([CGS04]) *La surface S issue de l'algorithme des MC d'un objet discret V pour un seuil dans $[0, 1]$ a les propriétés suivantes :*

- S est une 2-variété combinatoire sans bord ;
- la discréttisation de GAUSS de S est exactement V .

En d'autres termes, S est une polyédration réversible de V . Un inconvénient important est le nombre de facettes issues des MC (voir figure 6.11). Dans la section 6.3.2, nous allons simplifier cette surface en intégrant une information issue des plans discrets reconnus à la surface de V .

Pour terminer sur les liens entre la surface MC et l'objet discret, nous pouvons remarquer que chaque sommet de S est placé sur un segment $[p, p \pm \vec{d}]$ pour $p \in \mathbb{Z}^3$ avec

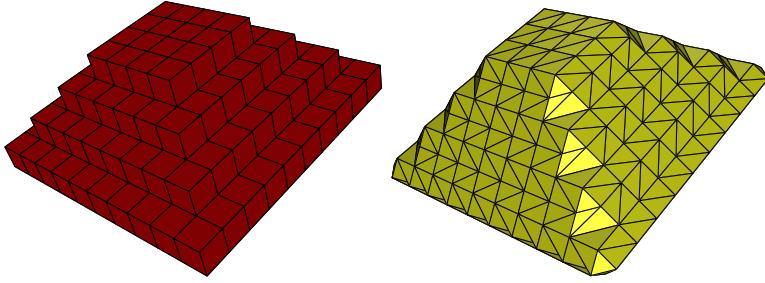


FIGURE 6.11 – Objet discret 3D et une surface MC associée.

$\vec{d} \in \{(0, 0, 1), (0, 1, 0), (1, 0, 0)\}$. Si le seuil s est tel que $0 < s < 1$, le segment précédent est ouvert. Ainsi, chaque sommet est associé à un surfel de la surface discrète (voir figure 3.9). Nous avons donc le corollaire suivant :

Corollaire 6.1 *Un déplacement d'un sommet s de S sur le segment ouvert $[p, p \pm \vec{d}]$ associé ne change pas les propriétés du lemme 6.1.*

6.3.1.2 Segmentation d'une surface discrète en plans discrets maximaux

Pour pouvoir inclure une information liée à la segmentation en plans discrets de la surface discrète dans la triangulation des MC, nous utilisons un algorithme d'étiquetage des surfels dérivé de [Siv04]. Nous ne détaillerons pas cette approche mais notons qu'à chaque morceau de plans discrets obtenu (ensemble de surfels connexes homéomorphe à un disque), nous avons la *préimage* complète associée (voir chapitre 5).

De manière intuitive, la préimage (polyèdre convexe dans l'espace des paramètres associé aux plans) définit l'ensemble des plans euclidiens dont la discréttisation (au sens de GAUSS) contient l'ensemble des surfels associé. Ainsi, nous avons :

LEMME 6.2 ([CGS04]) *Considérons un ensemble \mathcal{E} de surfels reconnu comme appartenant au plan discret P , alors tout plan euclidien dans la préimage de P intersecte les segments ouverts $[p, p + \vec{d}]$ associés aux surfels de \mathcal{E} .*

De manière intuitive, la propriété d'intersecter tous les segments ouverts $[p, p + \vec{d}]$ est à mettre en relation avec le processus de discréttisation (ici, de GAUSS) choisi pour la définition des plans discrets considérés. Rappelons qu'à ce niveau, l'étiquetage des surfels de la surface d'un objet en un nombre minimal de morceaux de plan discret est un problème NP-complet [SC06] dans le cas général.

Par l'association entre un sommet du MC et un surfel discret, nous fusionnons les informations en affectant l'étiquette d'un surfel à son sommet du MC associé. Nous définissons ainsi trois types de triangles des MC (voir figure 6.12) :

- **Homogène** : les trois sommets du triangle appartiennent au même plan discret (*i.e.* ils ont la même étiquette) ;
- **Non-homogène (2-NH ou 3-NH)** : cas contraire avec une sous classification 2-NH ou 3-NH si le nombre d'étiquettes distinctes est de 2 ou 3.

Généralement, les processus de segmentation de surface discrète en morceaux de plans maximaux incluent des contraintes topologiques sur les morceaux en question de telle manière que l'ensemble des triangles homogènes de même label soit homéomorphe à un disque.

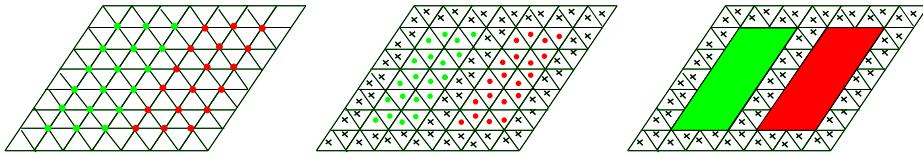


FIGURE 6.12 – Illustration du transfert des étiquettes des sommets vers les faces. Les triangles avec des croix sont 2-NH ou 3-NH.

6.3.2 Simplification et Optimisation

6.3.2.1 Triangles homogènes

Dans notre processus de simplification, la nature des triangles est primordiale. Dans le cas des triangles homogènes, d'après le lemme 6.2 et le corollaire 6.1, nous pouvons déplacer tous les sommets d'un ensemble connexe de triangles homogène sur un même plan euclidien extrait de manière arbitraire dans la préimage associée, et ce sans changer les propriétés de réversibilité et de correction topologique. Le traitement des triangles homogènes est donc très simple.

Dans ce contexte, nous pouvons ensuite remplacer tous les morceaux connexes de triangles homogènes par des polygones, *cousus* entre eux par des triangles 2-NH et 3-NH (voir figure 6.13). A ce niveau de l'analyse nous avons déjà un processus de polyédration réversible topologiquement correcte plus efficace que le MC en termes de nombre de facettes [CGS04].

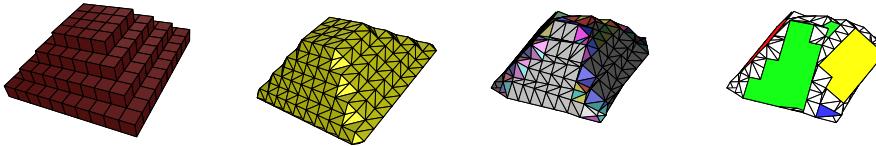


FIGURE 6.13 – Illustration du processus de simplification des triangles homogènes : (*de gauche à droite*) un objet discret, une surface des MC, segmentation de la surface en morceaux de plans (étiquettes ramenées aux triangles) et simplification des ensembles connexes de triangles homogènes.

Nous insistons sur le fait suivant : la réversibilité topologiquement correcte est donnée quel que soit le choix du représentant euclidien extrait des préimages des plans discrets.

6.3.2.2 Triangles non-homogènes

Le traitement des triangles non-homogènes va cette fois exploiter de manière fine les préimages associées. Si nous ajoutons des contraintes linéaires dans l'espace des paramètres réduisant la préimage, les points du polyèdre convexe résultant permettront toujours de simplifier les triangles homogènes comme énoncé dans la section précédente. L'idée est donc de trouver des contraintes linéaires pour supprimer des triangles NH, en plus des triangles homogènes.

Pour simplifier le problème et limiter le coût algorithmique, nous considérons deux restrictions au problème :

- **Analyse locale** : comme illustré dans la figure 6.14, la distance (au sens de HAUSDORFF) entre deux reconstructions valides pour la discrétilisation de GAUSS peut être très grande (et même arbitrairement grande). Dans ce qui suit, nous limiterons la

reconstruction à une zone définie comme l'union des cellules du MC.

- **Programmation linéaire en dimension 3** : comme nous l'avons vu dans le chapitre 5, le calcul des préimages utilise des outils de programmation linéaire en dimension 3. Dans ce qui suit, nous ne voulons pas augmenter la dimensionnalité du problème et donc des outils.

En considérant ces heuristiques, la simplification d'un triangle NH noté T , tout en maintenant la topologie (voir figure 6.15), peut se faire de deux manières :

- suppression d'une arête de T : dans ce cas nous allons nous restreindre aux arêtes contenues dans une face de la cellule de MC contenant T . Dans ce cas, l'arête est remplacée par un sommet contenu dans la face de la cellule et nous nous ramenons donc à un processus bidimensionnel (voir figure 6.16) ;
- suppression directe du triangle : T sera remplacé par un sommet et nous contraignons ce dernier à rester dans la cellule du MC associée à T . Ce schéma de simplification sera utilisé uniquement pour les 3-NH.

Supposons maintenant un triangle T 2-NH associé aux plans discrets P_1 et P_2 . Nous cherchons à contraindre les préimages de P_1 et P_2 tel que le déplacement des sommets de T sur n'importe quel plan euclidien extrait des préimages fusionnera une arête de T (et donc supprimera T). Pour cela, nous prenons l'exemple de la figure 6.17 : la figure (a) correspond aux configurations observables sur les faces d'une cellule du MC engendrant une seule composante connexe de triangles (cas 1, 2, 5, 8, 9, 11). Pour ne pas complexifier le processus, nous nous restreignons à ces configurations. Sur ce schéma, nous cherchons à contraindre la famille des plans "bleus" et la famille des plans "verts" de telle manière que l'intersection d'un représentant bleu et d'un représentant vert reste dans la face $ABCD$ (figure 6.17-(b)). Si nous exprimions ces contraintes de manière directe, nous aboutirions à un problème linéaire en dimension supérieure à 3 (3 paramètres par préimage).

Encore une fois nous allons simplifier le problème et construire des schémas nécessaires mais non suffisants : pour le cas de la figure 6.17-(b), si nous contraignons la famille P_1 à intersecter le segment DC et que nous contraignons (indépendamment cette fois) la famille P_2 à intersecter CB , alors nécessairement l'intersection $p_1 \cap p_2 \in ABCD$, quels que soient $p_1 \in P_1$ et $p_2 \in P_2$. Les contraintes sur les préimages P_1 et P_2 sont donc de dimension 3 et s'expriment directement en fonction des coordonnées du point A . Si maintenant les contraintes linéaires que nous ajoutons (en nombre constant par triangle) aux préimages P_1 et P_2 font qu'une des deux devient nulle, nous en concluons que le triangle T ne peut pas être supprimé. Comme le schéma est nécessaire mais non suffisant, il serait faux de conclure qu'il n'existe pas de couple (p_1, p_2) permettant de supprimer T après projection. Cependant, le calcul d'un tel couple ne pourrait se faire avec un système linéaire de dimension 3. Notons que la suppression de T ne changera ni la réversibilité totale de la surface, ni sa topologie (figure 6.15).

Nous ne détaillerons pas plus en avant ces contraintes mais nous pouvons construire des schémas équivalents pour les autres cas de la figure 6.17-(a) [CDJS06].

6.3.3 Algorithme global de simplification

Maintenant que nous savons contraindre les préimages pour supprimer un triangle 2-NH, il nous faut appliquer ce processus sur la triangulation du MC. Là encore, il nous faut faire un choix de parcours (et donc de propagation des contraintes). Pour maximiser le taux de triangles NH supprimé, nous avons opté pour un processus en deux étapes :

- **Étape 1 : optimisation globale** : nous choisissons un triangle 2-NH, nous contraignons ses préimages et ensuite nous continuons, tant que possible (*i.e.* tant que les deux préimages sont non-vides) sur les triangles 2-NH voisins associés aux mêmes plans. Ce processus est itéré pour tous les triangles 2-NH de la surface. A la fin de ce processus, nous avons, en contrignant les préimages, potentiellement supprimé un certain nombre de triangles 2-NH.

- **Étape 2 : optimisation gloutonne** : nous prenons une préimage et nous choisissons arbitrairement un plan euclidien. Ensuite, nous propagons ce choix à tous les triangles 2-NH et 3-NH adjacents à ce plan qui n'ont pas encore été marqués comme simplifiables. Durant cette propagation, nous allons pouvoir générer à nouveau des contraintes linéaires similaires au traitement du cas 6.17-(b) mais pouvant traiter aussi les triangles 3-NH car un des trois plans est maintenant fixe (voir [CDJS06] pour plus de détails). Nous itérons sur cette sélection et cette propagation du plan euclidien pour chaque préimage.

Une fois tous les plans euclidiens sélectionnés, nous effectuons la simplification finale de la surface en projetant les sommets sur les plans euclidiens considérés et en simplifiant les triangles exactement coplanaires (cas des triangles homogènes). Comme à chaque étape, nous avons contraint notre géométrie à rester dans les cellules du MC, et comme nous pouvons aisément modifier localement la géométrie et la topologie de la surface, nous pouvons conclure :

LEMME 6.3 ([CDJS06]) *L'algorithme précédent construit un polyèdre réversible qui est une 2-variété combinatoire sans bord.*

A ce niveau, les facettes ne sont pas triangulées et donc des étapes de post-traitement peuvent être envisagées en fonction de l'application.

6.3.4 Résultats et discussion

Nous présentons ici quelques résultats en utilisant l'algorithme de SIVIGNON pour la segmentation [SDC03] et CASSOWARY¹, une bibliothèque de programmation linéaire pour l'ajout de contraintes sur les préimages. D'un point de vue algorithmique, une fois les préimages calculées, le processus ajoute un nombre constant de contraintes par triangle NH et parcourt un nombre constant de fois la triangulation. Le point coûteux est donc le calcul des préimages et la mise à jour de ces dernières. Le chapitre 5 a présenté plus en détail la complexité de la préimage associée à un plan discret en terme de nombre de facettes.

La figure 6.18 et le tableau 6.1 synthétisent quelques résultats sur des objets simples. Ce qu'il faut cependant noter est que les choix heuristiques et les conditions non suffisantes dans certains processus permettent quand même d'avoir un taux de suppression des triangles NH très élevé.

La contre-partie de ces choix heuristiques, surtout dans le mode de parcours pour la segmentation en plans discrets ainsi que les modes de propagation locale des contraintes lors de la simplification des triangles, fait que la reconstruction de la sphère n'est pas isotrope. Ce phénomène a déjà été largement observé [Siv04, Dex06] et semble inhérent à ces schémas de segmentation locale de la surface.

Objets	MC			Taux de suppression	
	# triangles MC	# triangles H	# triangles NH	triangles NH	total
pyramid_4	512	342	170	62%	87%
rd_cube_7	2024	1720	304	89%	98%
sphere_10	3656	2200	1456	37%	75%

TABLE 6.1 – Quelques résultats quantitatifs sur les exemples de la figure 6.18.

6.4 Conclusion

Dans ce chapitre, nous nous sommes intéressés à des notions de modélisation d'objets euclidiens dans le modèle discret, ainsi qu'au processus inverse qui consiste à obtenir une

1. <http://www.cs.washington.edu/research/constraints/cassowary/>

représentation en compréhension d'un objet décrit en extension.

Nous avons présenté des outils de reconstruction réversible de courbes et de surfaces polygonales en dimension 2 et 3. La solution proposée en dimension 2 est toutefois plus générale car elle est utilisable pour toutes les grilles isothétiques irrégulières présentées dans le chapitre 3.

Pour la dimension 3, nous sommes allés au maximum d'un point de vue théorique sur la construction d'un polyèdre réversible par optimisation d'une surface initiale (MC) en intégrant des informations géométriques propres à la géométrie discrète. Si nous voulons limiter la dimensionnalité du problème et des algorithmes, nous avons vu qu'il fallait s'orienter vers des processus heuristiques et des critères non nécessaires et suffisants. Bien évidemment, il pourrait être intéressant de tenir compte des configurations du MC que nous n'avons pas encore traitées. Cela permettrait sûrement d'enlever encore certains triangles mais il nous semble cependant plus prometteur de revoir le processus de segmentation en plans discrets et de propagation pour rendre la reconstruction plus *globale*.

Pour cela, une généralisation en dimension 3 de la couverture tangentielle qui possède de nombreuses utilisations en dimension 2 [FT99, dV07] nous semble intéressante : pour chaque surfel de la surface discrète, nous construisons le plan discret maximal avec recouvrement. Ainsi, pour chaque surfel, nous avons une liste de plans discrets le contenant. Si pour l'instant la combinatoire de cet objet rend le coût algorithmique important, il contient une information globale sur laquelle on pourrait envisager une segmentation en plans, cette fois sans recouvrement, avec de meilleures propriétés géométriques.

Enfin, nous pourrions nous attacher à la généralisation la reconstruction 3D aux objets isothétiques irréguliers. Dans un premier temps, nous pouvons considérer des surfaces simples et des grilles issues de subdivisions régulières comme l'*octree* pour lequel des algorithmes d'extraction d'isosurfaces existent [NY06].

Quelle que soit la dimension, 2 ou 3, du problème, la reconstruction réversible est souvent mise en parallèle d'approches venant, soit de la vectorisation de documents dans le cas 2D, soit de la simplification de maillages triangulés pour les surfaces. Pour ces deux domaines de l'analyse d'images ou de la modélisation, de nombreuses techniques existent. S'il serait pertinent de vouloir se comparer à ces techniques en terme de qualité de la représentation, rappelons que notre contrainte initiale de réversibilité et donc du contrôle strict de l'erreur dans une certaine mesure, n'apparaît que rarement dans ces autres approches.

De ce constat, les challenges à venir pour ces processus de reconstruction sont sur ce terrain là : poursuivre l'analyse des algorithmes de reconstruction exacte en cherchant à les optimiser en terme d'arithmétique et d'algorithmique pour offrir des solutions alternatives à des problématiques nécessitant ou non un contrôle exact de l'erreur.

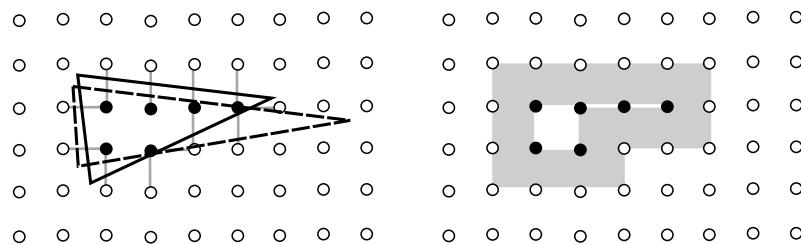


FIGURE 6.14 – (gauche) : Deux reconstructions possibles validant la propriété de réversibilité.
(droite) : nous restreignons la reconstruction à être contenue dans la zone grise.

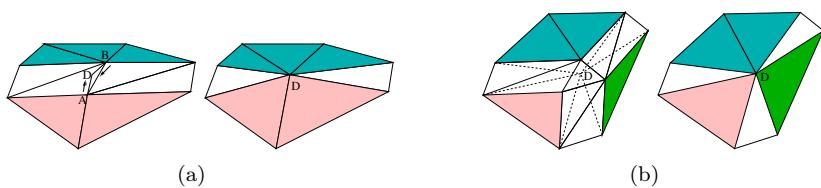


FIGURE 6.15 – Illustration de la suppression d'une arête et de la suppression d'un triangle.

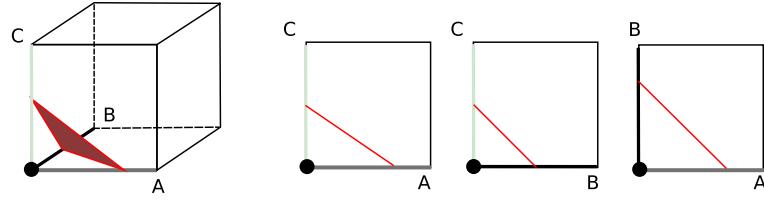


FIGURE 6.16 – Décomposition bidimensionnelle d'une cellule de MC et faces associées à un triangle.

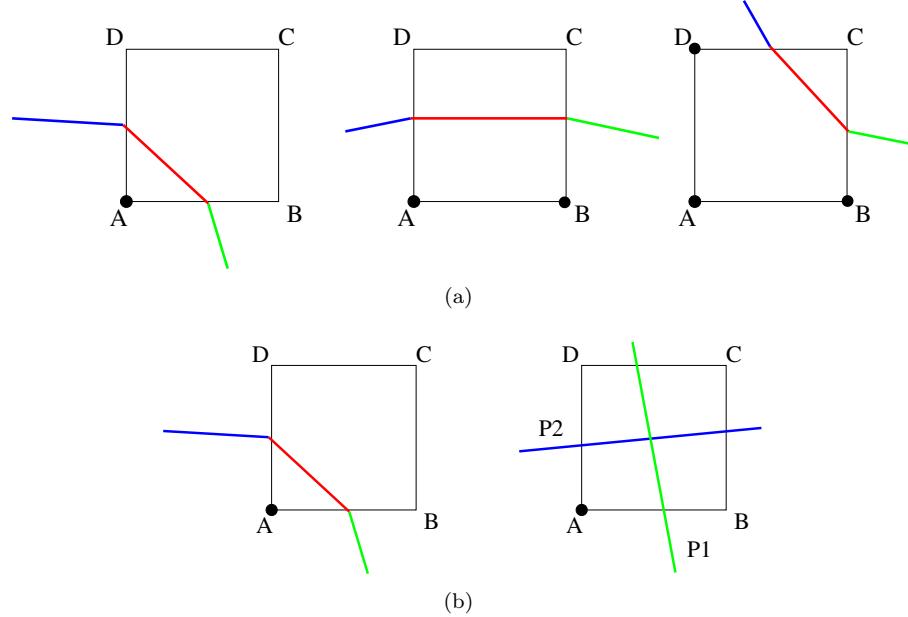


FIGURE 6.17 – Illustration du traitement en dimension 2 de la suppression d'un triangle T sur les faces de la cellule du MC associée.

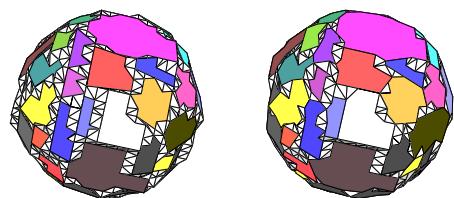


FIGURE 6.18 – Quelques résultats de la simplification : (*de gauche à droite*) les objets discrets, la surface MC, première simplification des triangles homogènes uniquement et enfin, simplification finale H+NH.

Chapitre 7

Algorithmes séparables pour l'analyse volumique de formes

Sommaire

7.1	Introduction	84
7.2	État de l'art	84
7.3	Transformation en distance, reconstruction et axe médian discret	85
7.3.1	Transformation en distance	85
7.3.2	Diagramme de VORONOÏ discret	86
7.3.3	Transformation en distance inverse	95
7.3.4	Extraction de l'axe médian	96
7.3.5	Liens avec la géométrie algorithmique	100
7.4	Généralisation aux espaces toriques	101
7.5	Conclusion	104

7.1 Introduction

Dans ce chapitre, nous nous intéressons à l'analyse volumique des objets discrets. Dans ce contexte la problématique de transformation en distance et de calcul d'axe médian sont des thématiques très importantes en géométrie discrète et elles bénéficient d'une longue histoire.

L'axe médian a trouvé de nombreuses applications en analyse d'images pour la reconnaissance et la description de formes [SHB99], ou en robotique pour la planification de trajectoires [Lat91]. La distance à la frontière, mémorisée en chaque point de l'axe médian, fournit une information d'épaisseur qui peut être utilisée pour segmenter les objets, les séparant en larges régions connectées par des parties relativement plus étroites [CPW93, DGG03, Ogn94, SS02]. La transformée en distance et l'axe médian peuvent aussi servir pour construire une représentation implicite du contour discret et ainsi permettre des rendus par lancer de rayon intéressants [CMC07, chap. 18].

Dans ce chapitre, nous insistons tout d'abord sur le caractère historique de ce domaine en dressant un état de l'art. Ensuite, nous présentons les différents algorithmes séparables pour le calcul de la transformation en distance, pour le calcul du diagramme de VORONOÏ discret, pour la transformation en distance inverse et enfin pour l'extraction de l'axe médian discret.

Par la suite, nous exploitons le caractère séparable des algorithmes pour présenter une généralisation de ces derniers aux espaces toriques.

La formulation de certains points, notamment dans la section 7.3.4, est issue de discussions avec DOMINIQUE ATTALI et ERIC REMY à l'occasion de la rédaction du chapitre [CMC07, chap. 9].

7.2 État de l'art

Dès les premières heures de la géométrie discrète, la transformation en distance et l'extraction d'axe médian de formes sont apparues comme des problématiques fondamentales avec de nombreuses applications [RP66, RP68]. Comme introduite dans la section 3.4.3, la transformation en distance consiste à étiqueter les points de l'objet discret par la distance minimale aux points du complémentaire.

Dans ce processus, nous avons un choix sur la distance discrète considérée. Ainsi, nous pouvons utiliser des distances approchées de chanfrein [RP68, Bor86, RT00a, FM05] ou des séquences de distances de chanfrein [RP66, MDKC00, Nag05], la distance euclidienne exacte basée sur un vecteur de déplacement [Dan80, Rag90, Mul92, CM99], la distance euclidienne donnée par un diagramme de VORONOÏ des points du complémentaire [BGKW95, GL95, GM98, MQR03], ou encore en manipulant le carré de la distance euclidienne [ST94a, Hir96, MRH00].

Historiquement, le choix de la distance sous-jacente était un compromis entre qualité de la représentation (par rapport à l'anisotropie notamment) et efficacité algorithmique. Dans ce contexte, les approches basées sur les masques de chanfrein offraient un avantage algorithmique certain. Les premières solutions efficaces en distance euclidienne souffraient cependant d'erreurs dans l'étiquetage en distance rendant les calculs inexacts [Dan80, Rag90, Mul92]. A présent, des solutions algorithmiques optimales en temps (linéaire au nombre de points discrets) existent pour un étiquetage en distance sans erreur en dimension arbitraire [BGKW95, GM98, Hir96, MRH00, MQR03]. La généralisation aux dimensions supérieures de ces approches est triviale car elle se basent sur un processus séparable opéré dimension par dimension. C'est sur ce type de technique que nous portons notre analyse.

Le squelette et l'axe médian sont des représentations de formes importantes très utilisés dans un objectif de reconnaissance de formes [Blu67, Mon69]. Plusieurs approches existent pour caractériser ce squelette ou cet axe médian. Par exemple, on peut considérer le modèle

de « feu de prairie » avec un front de propagation partant des contours. Les points d'auto-intersection définissent le squelette et la classification de ces derniers peut être obtenue en s'intéressant aux intersections et transitions durant ce processus [GK03, GK04]. Une autre approche consiste à s'intéresser aux pics et crêtes de la transformée en distance vue comme une surface. Enfin une dernière approche définit l'axe médian comme les centres des boules maximales contenues dans la forme. Nous reviendrons plus en détail sur cette définition dans la section 7.3.4.

De ces définitions souvent équivalentes dans l'espace continu ont découlé plusieurs objets différents dans l'espace discret. La détection des pics et crêtes a introduit les approches variationnelles [LL92, KSKB95, SBTZ99, CTK00] ; du modèle de « feu de prairie » ont découlé les approches par amincissement progressif et d'où la notion de squelette topologique discret (voir [CMC07, chap. 8] pour une bibliographie complète). Enfin les approches par boules maximales ont donné naissance à la notion d'axe médian discret que nous utiliserons par la suite, basée sur une distance de chanfrein [RP66, Bor86, dB94, RT02] ou sur une distance euclidienne [Rag93, GF96, ST94b, RT05, CM07].

Nous allons commencer dans la section suivante par présenter les différents algorithmes séparables pour les problèmes de transformation en distance, de l'extraction d'un diagramme de VORONOÏ discret, de transformation inverse et d'extraction d'axe médian. Dans la section 7.4, nous exploiterons le caractère séparable des algorithmes pour présenter quelques généralisations.

7.3 Transformation en distance, reconstruction et axe médian discret

7.3.1 Transformation en distance

Pour définir des algorithmes corrects, il a fallu revoir le processus avec des techniques utilisant des algorithmes séparables, c'est-à-dire des processus qui effectuent des calculs sur les lignes, puis sur les colonnes (puis sur les rangées en dimension 3, etc) de l'image de manière indépendante [ST94a]. Pour représenter la distance de manière exacte, nous utilisons le carré de cette dernière.

La généralisation de ce qui suit est triviale en dimension supérieure mais pour plus de simplicité dans la rédaction, nous considérons le cas de la dimension 2. Supposons une forme discrète X contenue dans une image de taille $m \times m$. Nous cherchons à calculer l'image $H = \{h(i, j)\}$ contenant le carré de la transformée en distance. De manière très simple, nous avons :

$$h(i, j) = \min \left\{ (i - x)^2 + (j - y)^2 : 0 \leq x, y < m \text{ et } (x, y) \in \overline{X} \right\}.$$

Cette écriture peut se décomposer dimension par dimension (voir figure 7.1) :

1. construction de l'image $G = \{g(i, j)\}$ par un traitement indépendant des lignes, où pour une colonne j nous avons :

$$g(i, j) = \min_x \left\{ |i - x| : 0 \leq x < m \text{ et } (x, j) \in \overline{X} \right\} ;$$

2. enfin, H s'obtient par le processus suivant sur les colonnes :

$$h(i, j) = \min_y \left\{ g(i, y)^2 + (j - y)^2 : 0 \leq y < m \right\} .$$

L'algorithme issu de cette décomposition engendre une complexité linéaire au nombre de points de l'image. En effet, l'étape 1 (voir algorithme 2) s'exécute en 2 parcours de l'image contenant X , donc en $O(m^2)$, et $O(m^n)$ en dimension n . Concernant la phase 2 (voir algorithme 3), la complexité est la même, mais l'algorithme requiert une analyse plus

précise. A partir du calcul de l'enveloppe inférieure d'une famille de paraboles : considérons la colonne $\{g(i, y)\}$ ($0 \leq y < m$) de l'image G de la phase 1. Si nous considérons l'ensemble des paraboles $\mathcal{F}_y^i(j) = g(i, y)^2 + (j - y)^2$, la colonne $\{h(i, y)\}$ après l'étape 2 est exactement la hauteur de l'enveloppe inférieure des paraboles $\{\mathcal{F}_y^i\}$ pour $0 \leq y < m$ (voir figure 7.2).

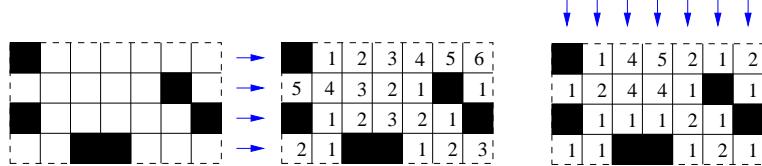


FIGURE 7.1 – Illustration du calcul de la transformée en distance de manière séparable (dimension par dimension).

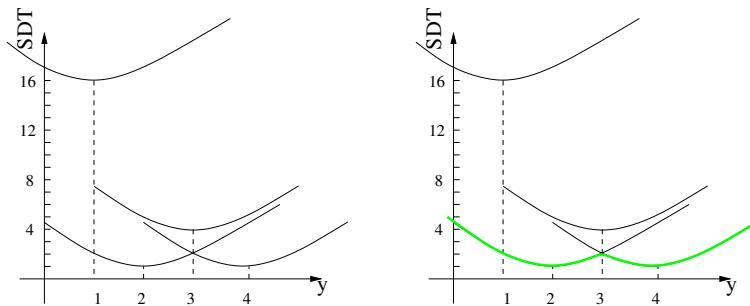


FIGURE 7.2 – Illustration du calcul de l'enveloppe inférieure de paraboles : si $[16, 1, 4, 1]$ est une colonne de G après l'étape 1 (5^{ème} colonne dans la figure 7.1, valeurs élevées au carré), on obtient (à gauche) l'ensemble des paraboles $g(i, y)^2 + (j - y)^2$ et (à droite) la courbe épaisse correspondant à l'enveloppe inférieure ; le résultat est $[2, 1, 2, 1]$.

Pour calculer l'enveloppe inférieure d'une famille de paraboles, un algorithme existe pour rendre l'étape 2 linéaire au nombre de points discrets [Hir96]. L'idée de cet algorithme (détailé dans l'algorithme 3, inspiré de [MRH00]) est de manipuler une pile pour mémoriser les paraboles présentes dans l'enveloppe inférieure en cours d'extraction. Ensuite, à chaque fois qu'une parabole est analysée, nous pouvons avoir à dépiler des paraboles de la pile. La complexité linéaire est obtenue par le fait que les paraboles ne sont considérées qu'une seule fois et que lorsqu'une parabole est dépiler, elle est supprimée du problème (elle ne fera pas partie de l'enveloppe). Deux fonctions sont utilisées dans cet algorithme : $\mathcal{F}_y^i(j) = g(i, y)^2 + (j - y)^2$ (ou plus simplement $\mathcal{F}_y(j)$ lorsque i est donné par le contexte) représentant l'ordonnée d'une parabole, et la fonction $Sep^i(u, v)$ (ou simplement $Sep(u, v)$) représentant l'abscisse de l'intersection de deux paraboles consécutives :

$$Sep(u, v) = (v^2 - u^2 + g(i, v)^2 - g(i, u)^2) \operatorname{div} (2(v - u))$$

La figure 7.4 présente deux représentations classiques de la transformée en distance : sous forme de fonction de hauteur ou avec une dynamique de niveaux de gris cyclique réduite pour mieux observer les fronts de propagation. En dimension 3 (voir figure 7.5), la visualisation peut se faire par coupe dans le volume.

7.3.2 Diagramme de VORONOÏ discret

Pour certaines applications, il est parfois nécessaire de mémoriser en chaque point de X , non seulement sa distance au complémentaire, mais aussi le ou les points discrets du complémentaire les plus proches. Une telle décomposition est appelée *feature transform*,

Algorithme 2 : Phase 1 de SEDT.

Données : X forme binaire dans une image $m \times m$
Résultat : L'image g contenant l'EDT selon l'axe des x

```

pour  $j \in [0..m - 1]$  faire
    si  $(0, j) \in \bar{X}$  alors
         $| g(0, j) := 0;$ 
    sinon
         $\| g(0, j) := \infty;$ 
    pour  $i := 1$  à  $m - 1$  faire
        si  $(i, j) \in \bar{X}$  alors
             $| g(i, j) := 0;$ 
        sinon
             $\| g(i, j) := 1 + g(i - 1, j);$ 
    pour  $i := m - 2$  à  $0$  faire
        si  $g(i + 1, j) < g(i, j)$ 
        alors
             $\| g(i, j) := 1 + g(i + 1, j);$ 

```

Algorithme 3 : Phase 2 de SEDT.

Données : L'image de distance g donnée à la phase 1 de l'algorithme
Résultat : L'image h des valeurs de la SEDT

```

pour  $i \in [0..m - 1]$  faire
     $q := 0; s[0] := 0; t[0] := 0;$ 
    pour  $j := 1$  à  $m - 1$  faire
        tant que  $(q \geq 0)$  et  $(\mathcal{F}_{s[q]}(t[q]) > \mathcal{F}_j(t[q]))$  faire
             $\| q := q - 1;$ 
        si  $q < 0$  alors
             $\| q := 0; s[0] := j;$ 
        sinon
             $\| w := 1 + \text{Sep}(s[q], j);$ 
            si  $w < m$  alors
                 $\| q := q + 1; s[q] := j;$ 
                 $\| t[q] := w;$ 
    pour  $j := m - 1$  à  $0$  faire
         $h(i, j) := \mathcal{F}_{s[q]}(j);$ 
        si  $j = t[q]$  alors
             $\| q := q - 1;$ 

```

FIGURE 7.3 – Algorithmes pour le calcul de la transformée en distance euclidienne d'un objet discret en dimension 2.

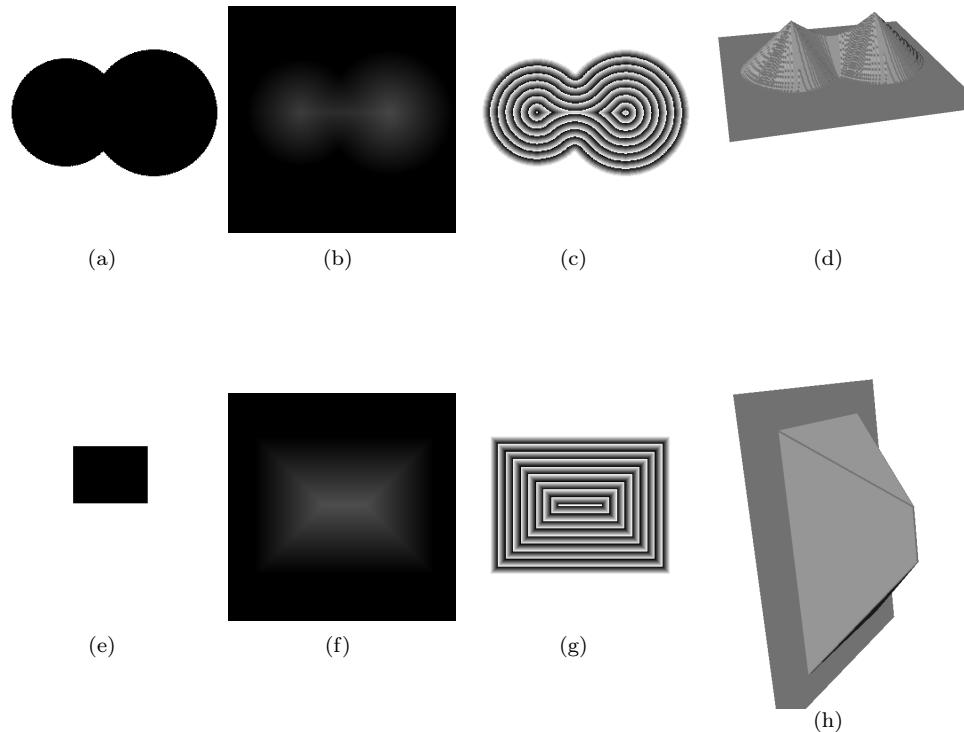


FIGURE 7.4 – Exemples de transformées en distance en dimension 2 avec différentes représentations : objets binaires, valeur de la transformée en distance, représentation des valeurs sur une plage réduite de niveaux de gris et représentation surfacique (la hauteur correspond à la valeur de la transformée).

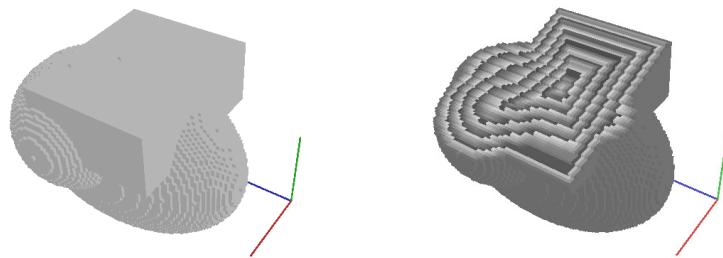


FIGURE 7.5 – Représentation d'une transformée en distance euclidienne 3D par coupe dans le volume (ici avec une palette circulaire de niveaux de gris).

tessellation de DIRICHLET [Bor86] ou encore fonction bissectrice [CCZ07]. Remarquons que cette transformation correspond exactement à l'intersection entre l'espace discret \mathbb{Z}^2 et le diagramme de VORONOÏ continu [PS85, dBvKOS00] des points du complémentaire.

Étant donné $S = \{s_i\}$ un ensemble de points ou *sites* de \mathbb{R}^2 , le diagramme de VORONOÏ de S est la partition du plan en cellules $C = \{c_i\}$ (une cellule par site s_i) telle que pour tout point p dans la cellule ouverte c_i , nous avons $d(p, s_i) < d(p, s_j)$ pour tout $j \neq i$. Nous définissons aussi $c_i^- = \{p \in \mathbb{R}^2; d(p, s_i) \leq d(p, s_j)\}$, la fermeture des cellules, le bord d'une cellule correspondant au lieu des points équidistants d'au moins deux sites.

Dans le plan discret, plusieurs algorithmes existent pour extraire un diagramme de VORONOÏ discret dans lequel on étiquette les points discrets par le label i si ceux-ci sont dans la cellule (ouverte) c_i [Coe02, MQR03, HVR05]. Pour ces techniques, si un point discret est exactement sur une frontière entre deux cellules, un choix arbitraire est fait concernant l'étiquetage. Notons que si le diagramme est utilisé dans un objectif de transformation en distance [Coe02, MQR03], ce choix n'a pas de conséquence.

Dans [CCZ07], nous avons présenté un algorithme complet dans lequel pour chaque point nous construisons *l'ensemble de projection de x sur S* : $\Pi_S(x) = \{y \in S, \forall z \in S, d(y, x) \leq d(z, x)\}$. Ainsi, $|\Pi_S(x)| = 1$ si x est dans une cellule ouverte du diagramme euclidien, et $|\Pi_S(x)| > 1$ si x est exactement sur un bord d'une cellule. En géométrie algorithmique classique, nous avons souvent une hypothèse de *position générale* pour les algorithmes spécifiant qu'il n'existe pas plus de trois points cocycliques. La conséquence est que les noeuds du diagramme résultant sont de degré trois et donc la triangulation de DELAUNAY peut être donnée sans ambiguïté.

En géométrie discrète, si nous construisons $\Pi_{\overline{X}}(p)$ pour tout $p \in \mathbb{Z}^2$, le nombre de points cocycliques pourra généralement être très élevé. Dans [CCZ07] l'intérêt de construire cette projection complète est que nous avons ainsi en chaque point une information globale non seulement sur la distance au bord, mais aussi sur le caractère “minimal local” de cette mesure. Cela rejoint la notion de flux contruit sur le diagramme de VORONOÏ classique [DGG03].

Grâce au caractère séparable des algorithmes de transformation en distance, nous pouvons construire très simplement cette projection. Tout d'abord, nous appelons *point multiple* tout point discret p tel que $|\Pi_{\overline{X}}(p)| > 1$. Concernant la première étape donnée dans l'algorithme 2, l'étiquetage consiste à mémoriser les coordonnées (ici l'abscisse) du ou des points pour lesquels la distance est minimale. Dans ce processus monodimensionnel, la détection des points multiples est triviale et $|\Pi_{\overline{X}}(p)| \in \{1, 2\}$ (voir figure 7.6-(a)). D'un point de vue structurel, nous manipulons, pour chaque point p de l'objet, une liste contenant des pointeurs vers les coordonnées des points de $\Pi_{\overline{X}}(p)$.

Pour l'étape 2 (algorithme 3), nous allons utiliser un processus équivalent aux techniques de construction du diagramme de VORONOÏ euclidien par balayage [PS85, dBvKOS00]. Étant donnée une colonne et en utilisant la vision “famille de paraboles” associée, certains événements peuvent être caractérisés lors de l'analyse du point p sur cette colonne. L'idée est la suivante :

- p est à l'intersection de deux paraboles. Dans ce cas, si p n'était pas point multiple à la dimension précédente, il le devient (cas $j = 4$ dans la figure 7.6) :
- si p était point multiple à la dimension précédente et que la parabole associée n'est plus dans l'enveloppe inférieure, p n'est plus point multiple (il n'est plus sur un bord du diagramme de VORONOÏ (cas $j = 7$).

Dans les autres cas, sont ajoutés aux pointeurs associés à p , les pointeurs associés au centre de la parabole de l'enveloppe inférieure “couvrant” p . Au cours de cette opération de *concaténation* des listes de pointeurs, nous ne propageons bien sûr que les sites de distance minimale. Par exemple, pour le cas $j = 4$, nous avons à fusionner les listes $\{K, L\}$ et $\{C, D\}$. Or, comme la parabole issue de la première liste n'est pas dans l'enveloppe inférieure, nous ne gardons que la seconde liste. Si maintenant la parabole issue de $\{K, L\}$ était préservée à un autre endroit dans l'enveloppe inférieure, nous aurions de toute façon conservé l'ensemble $\{C, D\}$ car il minimise la distance à l'enveloppe en $j = 4$. Si la hauteur de la parabole $\{K, L\}$ avait

été maintenant exactement la même que celle de $\{C, D\}$, nous aurions cette fois l'ensemble $\{C, D, K, L\}$ associé à p .

Dans l'implémentation actuelle, l'étape 1 se calcule en temps linéaire comme pour la transformée en distance. Pour l'étape 2, si les événements se détectent facilement sur l'enveloppe inférieure des paraboles, nous avons quelques opérations de concaténation avec suppression de doublons. Pour cela, nous avons opté pour une structure de donnée des listes non optimale mais plus simple à manipuler. Le coût de cet algorithme est en $O(m' + k)$ où m' est le nombre de points discrets de \bar{X} et $k = \sum_{p \in X} |\Pi_{\bar{X}}(p)|$.

Informellement, l'algorithme est donc linéaire en fonction du nombre de points discrets ainsi qu'à la cardinalité des points multiples. Notons que le coût en mémoire est en $O(k)$.

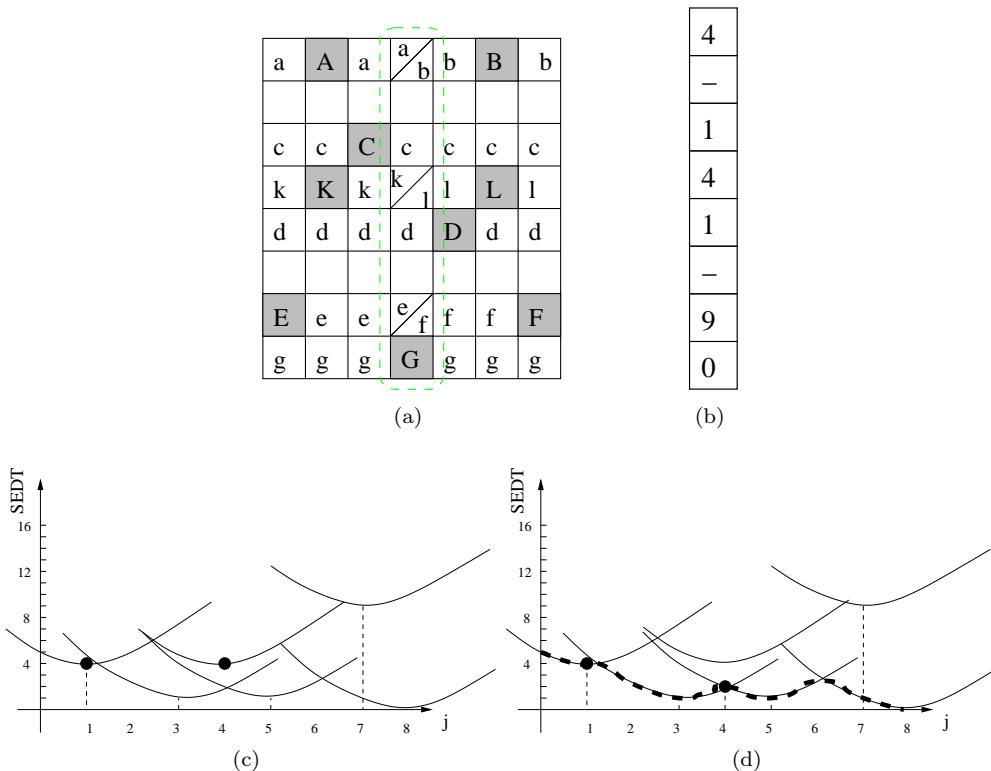


FIGURE 7.6 – Représentation de la décomposition en dimension pour le calcul du diagramme de VORONOÏ discret. (a) : Image en entrée (les pixels de \bar{X} sont en gris), et résultat de l'étiquetage selon l'axe des x . Les collisions sont indiquées par des doubles lettres. (b) : la valeur de transformée en distance de la colonne $i = 4$. (c) : l'ensemble des paraboles pour la colonne $i = 4$. Les points indiquent les collisions ($\#\Pi_{SP} > 1$) détectées dans la dimension précédente. (d) : l'enveloppe inférieure des paraboles et détection des nouveaux points tels que $\#\Pi_{SP} > 1$: le point $j = 7$ n'est plus sur une arête du diagramme, le point $j = 1$ est préservé et un nouveau point équidistant+ de plusieurs sites a été détecté $j = 4$.

Les figures 7.7 et 7.8 présentent quelques résultats de l'algorithme en dimension 2. Étant donné un objet discret, celui-ci retourne la transformée en distance ainsi que la structure complète $\Pi_{\bar{X}}$. Dans cette figure, nous utilisons une représentation sous forme de graphe orienté de la fonction $\Pi_{\bar{X}}$: uniquement les sommets de la forme sont représentés par des cercles et une arête orientée d'un sommet $p \in X$ vers un sommet $q \in \bar{X}$ indique que $q \in \Pi_{\bar{X}}(p)$. Un sommet de l'objet avec plus d'une arête incidente correspond à un point multiple dans le diagramme de VORONOÏ.

Bien évidemment, le processus de propagation des points multiples peut se généraliser

7.3. Transformation en distance, reconstruction et axe médian discret

en dimension supérieure. Néanmoins, la cardinalité de $\Pi_{\overline{X}}$ augmente et est à mettre en relation avec des problèmes fondamentaux de l'arithmétique comme le nombre de représentations d'un nombre par une somme de carrés [HW75]. Si le graphe $\Pi_{\overline{X}}$ ou l'étiquetage ne se représente pas très bien en dimension 3, nous avons cependant évalué dans [CCZ07] la distribution des points multiples sur certaines formes.

Tout d'abord, dans la figure 7.9, nous évaluons k (somme des $\Pi_{\overline{X}}$) par rapport à $m = |X|$. Ainsi, nous pouvons observer que le coût uniforme de l'algorithme est quasi linéaire, avec un sur-coût notable en dimension 3.

Si maintenant nous nous intéressons à la distribution des $|\Pi_{\overline{X}}|$, la figure 7.10 présente ces résultats pour des formes 2D et 3D.

En conclusion, nous obtenons par ces techniques le diagramme de VORONOÏ discret en extension, c'est-à-dire nous avons le résultat de l'intersection entre le diagramme de VORONOÏ continu des points de \overline{X} , et la grille \mathbb{Z}^n . Si nous voulons le graphe de VORONOÏ associé (c'est-à-dire une représentation cellulaire faces/arêtes/sommets), il nous faut un processus supplémentaire.

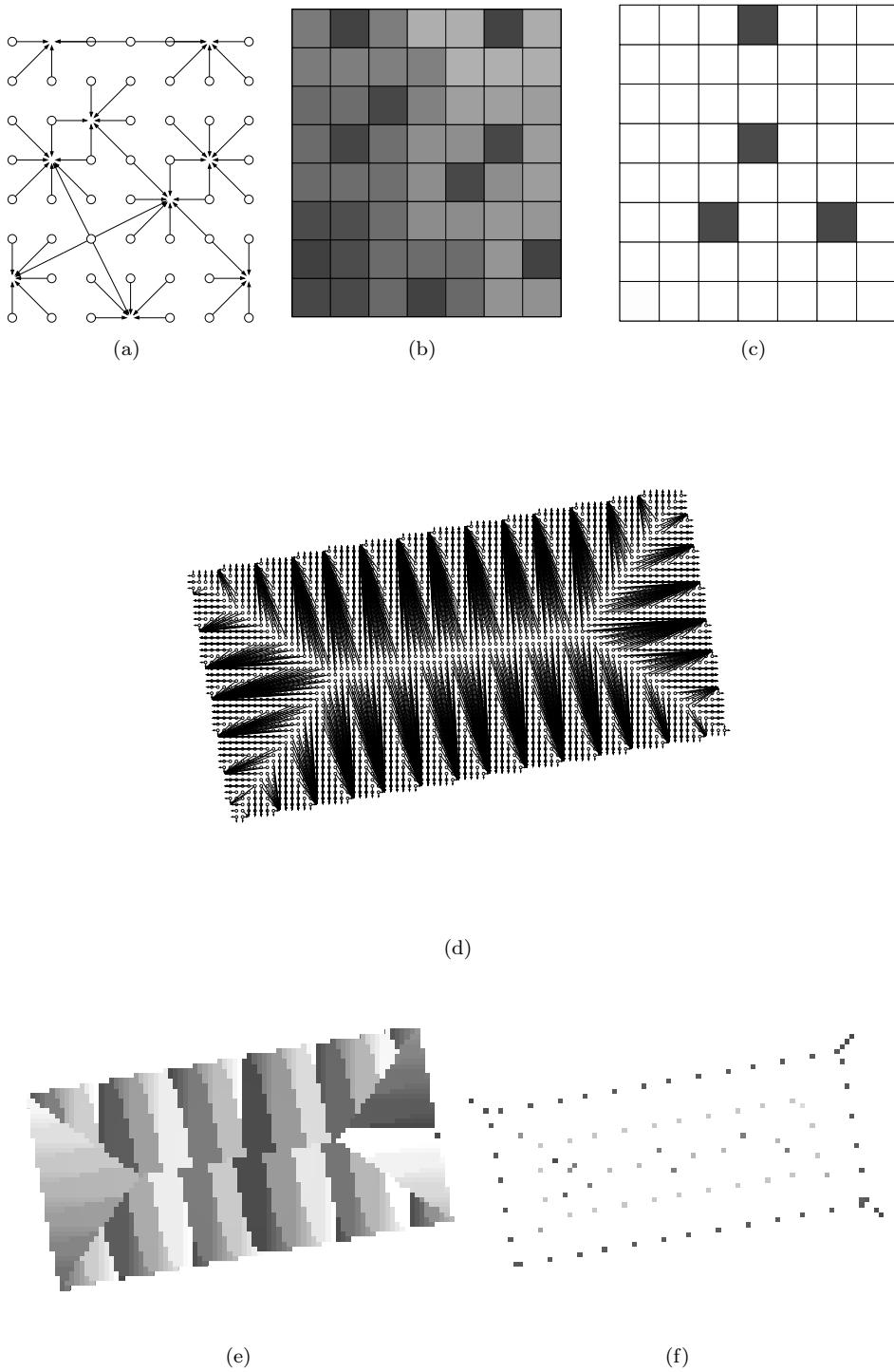


FIGURE 7.7 – Résultats obtenus par l'algorithme de calcul du diagramme de VORONOÏ discret sur l'objet de la figure 7.6 (a – c) et sur un rectangle désaxé (d – f). Nous traçons, l'étiquetage sous forme de graphe orienté de $\Pi_{\bar{X}}$ (a) et (e), d'étiquetage de VORONOÏ (affectation arbitraire sur les bords, (b) et (e)) et points discrets pour lesquels $\Pi_{\bar{X}}(p) > 1$ (c) et (f).

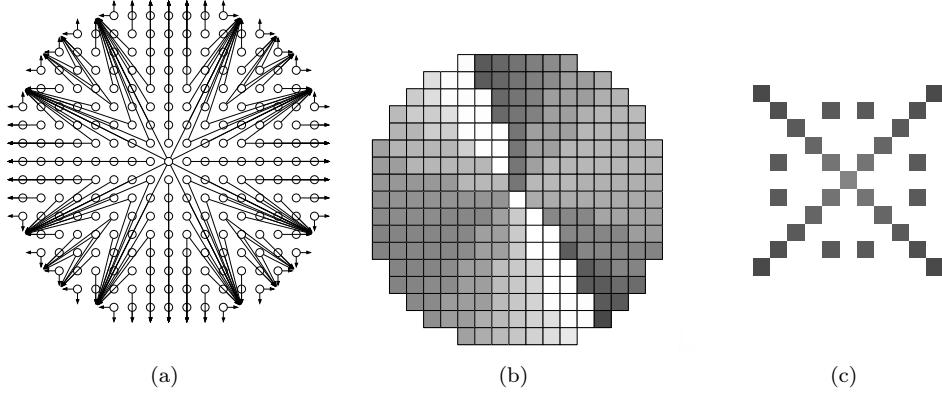


FIGURE 7.8 – Résultats sur un disque discret.

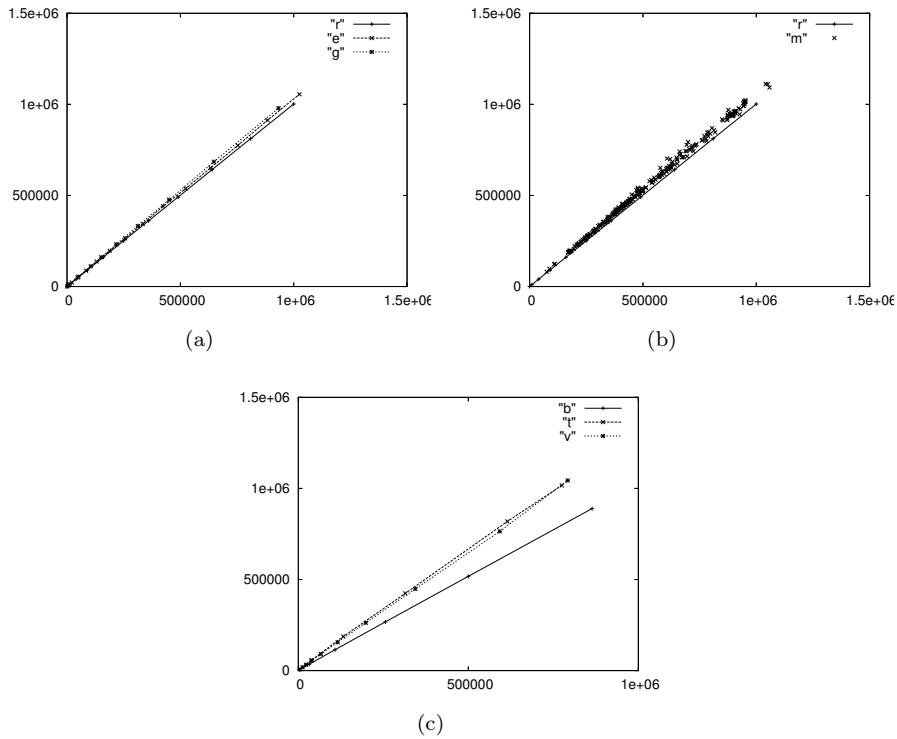


FIGURE 7.9 – Tracés de k en fonction de m (axe horizontal) : (a – b) formes 2D (rectangles “r”, ellipses “e”, images quelconques “m”), et (c) formes 3D (parallélépipèdes rectangles “b”, tores “t” et images de vertèbres 3D ”v”) [CCZ07].

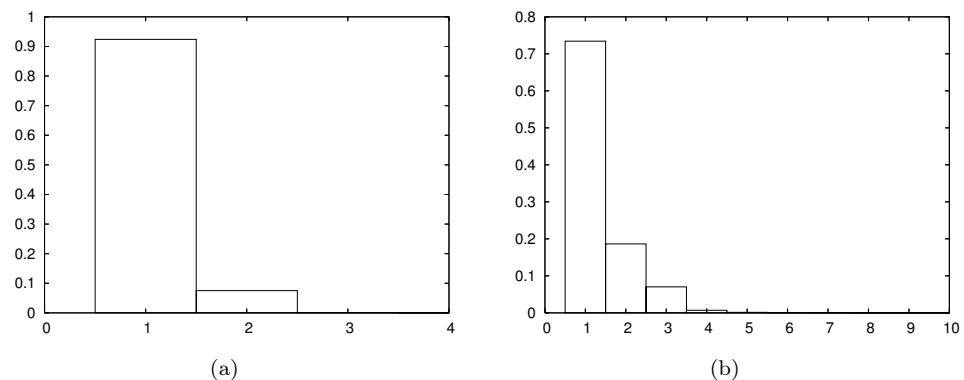


FIGURE 7.10 – Distribution de $|\Pi_{\bar{X}}(p)|$ sur une forme 2D (a) et sur une vertèbre 3D (b) [CCZ07].

7.3.3 Transformation en distance inverse

Avant de nous intéresser plus en avant à la notion d'axe médian, nous présentons un algorithme séparable et optimal en temps pour le problème de la transformation en distance inverse. Le problème en dimension n est le suivant : étant donné un ensemble de l points P_k de \mathbb{Z}^n associés à des rayons r_k , comment reconstruire efficacement l'objet $X = \bigcup_k B_d^<(P_k, r_k)$ où $B_d^<(P, r)$ est l'ensemble $\{Q \in \mathbb{Z}^n, d(P, Q) < r\}$.

Si nous reformulons :

$$X = \{Q \in \mathbb{Z}^n \mid \max_{1 \leq k \leq l} \{r_k^2 - \|Q - P_k\|^2\} > 0\}. \quad (7.1)$$

Pour obtenir un algorithme efficace, considérons l'image $f : \mathbb{Z}^n \rightarrow \mathbb{Z}$ obtenue en formant $f(P_k) = r_k^2$ et $f(Q) = 0$ pour $Q \notin \{P_1, \dots, P_l\}$ ainsi que l'image $h : \mathbb{Z}^n \rightarrow \mathbb{Z}$ telle que :

$$h(Q) = \max_{P \in \mathbb{Z}^n} \{f(P) - \|Q - P\|^2\}.$$

La forme X peut être retrouvée en conservant les valeurs strictement positives de h , $X = \{P \in \mathbb{Z}^n \mid h(P) > 0\}$. Ainsi, pour reconstruire X , il suffit de calculer h . Comme dans le cas d'une construction de la transformation en distance avec une distance euclidienne, nous décomposons ce processus de maximisation dimension par dimension (voir figure 7.11). Pour plus de clarté, nous considérons la reconstruction en dimension 2, la généralisation en dimension n se faisant aisément. En dimension 2, avec la notation $P = (x, y)$ des points discrets et en supposant que $X \subseteq [1, m]^2$, nous cherchons à calculer :

$$h(x, y) = \max_{1 \leq i, j \leq m} \{f(i, j) - (x - i)^2 - (y - j)^2\}. \quad (7.2)$$

Pour cela, nous procédons en deux étapes. Nous construisons tout d'abord l'image $g : [1, m]^2 \rightarrow \mathbb{Z}$ telle que :

$$g(x, y) = \max_{1 \leq i \leq m} \{f(i, y) - (x - i)^2\}.$$

L'image h est ensuite donnée à partir de g par :

$$h(x, y) = \max_{1 \leq j \leq m} \{g(x, j) - (y - j)^2\}.$$

Pour chacune de ces étapes, nous avons à calculer l'enveloppe supérieure d'une famille de paraboles, définies comme les graphes des fonctions $\{\mathcal{F}_i^y(x) = f(i, y) - (x - i)^2\}_{1 \leq i \leq m}$ et $\{\mathcal{G}_j^x(y) = g(x, j) - (y - j)^2\}_{1 \leq j \leq m}$. Or, dans la section 7.3.1, nous avons présenté un algorithme très similaire pour calculer une enveloppe de paraboles en $O(m)$. La seule différence est que les paraboles sont cette fois inversées et que la fonction de séparation est donnée par $Sep(u, v) = (u^2 - v^2 - f(u, y) + f(v, y)) \text{ div } (2(u - v))$ (voir figure 7.12).

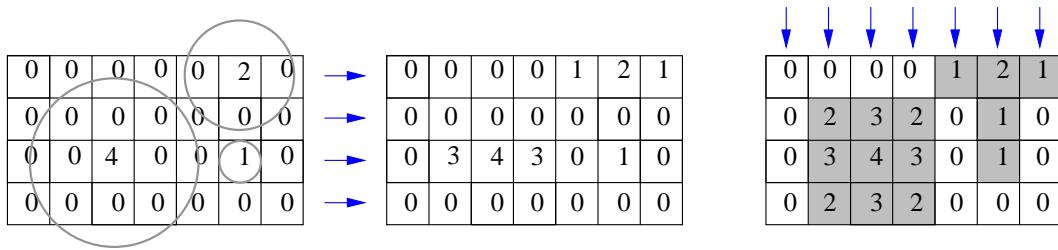


FIGURE 7.11 – Illustration du processus dimension par dimension pour la reconstruction en distance euclidienne.

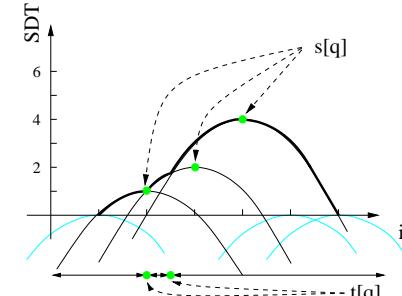
Finalement, en appliquant l'algorithme 4 pour toutes les dimensions, nous obtenons un algorithme de reconstruction dont la complexité est linéaire aux nombres de points discrets de la grille [CM07]. Pour une reconstruction avec une distance de chanfrein, voir [CMC07, chap. 9].

Algorithme 4 : Phase 1 de l'algorithme de REDT.

Données : la carte de distance f
Résultat : l'image h_0 des valeurs de la REDT

```

Pour  $j \in [1..m]$  Faire
     $q := 0$ ;  $s[0] := 0$ ;  $t[0] := 0$ ;
    Pour  $i := 2$  à  $m$  Faire
        Tant que  $(q \geq 0)$  et
             $(\mathcal{F}_{s[q]}(t[q]) < \mathcal{F}_j(t[q]))$  Faire
             $\quad q := q - 1$ ;
        Si  $q < 0$  alors
             $\quad q := 0$ ;  $s[0] := i$ ;
        Sinon
             $\quad w := 1 + \text{Sep}(s[q], j)$ ;
            Si  $w < n$  alors
                 $\quad q := q + 1$ ;  $s[q] := i$ ;
             $\quad t[q] := w$ ;
        Pour  $i := m$  à  $1$  Faire
             $h_0(i, j) := \mathcal{F}_{s[q]}(i)$ ;
            Si  $i = t[q]$  alors
                 $\quad q := q - 1$ ;
    
```



F	0	0	1	2	4	0	0
G	0	0	1	3	4	3	0
REDT	0	0	1	1	1	1	0

FIGURE 7.12 – Illustration du processus dimension par dimension pour la reconstruction en distance euclidienne.

7.3.4 Extraction de l'axe médian

Revenons sur les définitions initiales de l'axe médian dans le cas continu :

DÉFINITION 7.1 (Boule maximale) Une boule ouverte $B \subseteq X$ est maximale dans X si pour toute boule ouverte B' ,

$$B \subseteq B' \subseteq X \implies B = B'.$$

DÉFINITION 7.2 (Axe médian) L'axe médian, $\text{AM}(X)$, est l'ensemble des centres des boules maximales de X .

En géométrie algorithmique, le *squelette* est une notion voisine qui formalise l'image du feu de prairie de [Blu67]. Formellement, le *squelette* de X est l'ensemble des points qui ont au moins deux points les plus proches dans le complémentaire de X . Les notions de squelette et d'axe médian sont proches mais non équivalentes (voir figure 7.13). En effet, le squelette est contenu dans l'axe médian qui est lui-même contenu dans la fermeture du squelette [Mat88, chapitre 11]. Cependant, pour les unions finies de boules ouvertes, les deux notions coïncident.

En géométrie discrète, les notions de squelette et d'axe médian ne peuvent être mises en relation : si le squelette et l'axe médian continu sont homotopes à la forme originale [Lie04], seul le squelette discret s'attachera à cet objectif [CMC07, chap. 8]. L'axe médian discret est quant à lui purement géométrique.

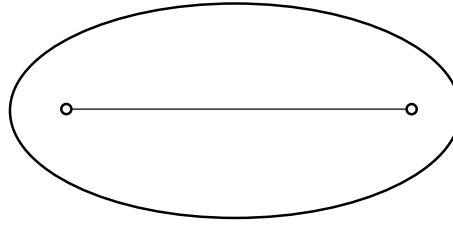


FIGURE 7.13 – Illustration de la différence entre axe médian et squelette dans le cas continu : l’arête fermée appartient à l’axe médian mais non au squelette qui ne contient que l’arête ouverte.

Nous commençons par donner une caractérisation des points de l’axe médian continu s’appuyant sur un relèvement des boules et des formes dans \mathbb{R}^3 . Par la suite, nous identifions les points (x, y) de \mathbb{R}^2 aux points $(x, y, 0)$ de \mathbb{R}^3 . Considérons une boule ouverte $B \subseteq \mathbb{R}^2$ de centre (i, j) et de rayon r et notons :

$$h_B(x, y) = r^2 - (x - i)^2 - (y - j)^2.$$

La boule B est l’ensemble des points (x, y) tels que $h_B(x, y) > 0$. Nous associons à B le paraboloïde elliptique \mathcal{P} de \mathbb{R}^3 défini comme l’ensemble des points $(x, y, z) \in \mathbb{R}^3$ d’équation $z = h_B(x, y)$. La relation géométrique entre la boule B et le paraboloïde \mathcal{P} est la suivante : les points en-dessous du paraboloïde \mathcal{P} intersectent le plan $z = 0$ en la boule B . Nous appelons *dôme* associé à B l’ensemble des points $(x, y, z) \in \mathbb{R}^3$ en-dessous du paraboloïde \mathcal{P} et au-dessus du plan $z = 0$:

$$\hat{B} = \{(x, y, z) \in \mathbb{R}^3 \mid 0 \leq z < h_B(x, y)\}.$$

La restriction du dôme au plan $z = 0$ redonne la boule B . Cet objet a été proposé pour la première fois dans [ST94b] pour caractériser les points de l’axe médian. Considérons une forme continue $Y \subseteq \mathbb{R}^2$. Si l’on substitue à l’ensemble des boules B incluses dans Y le dôme \hat{B} , on obtient la forme de \mathbb{R}^3 :

$$\hat{Y} = \bigcup_{B \subseteq Y} \hat{B}.$$

Nous dirons qu’un dôme est *maximal* dans \hat{Y} s’il n’est inclus dans aucun autre dôme contenu dans \hat{Y} . Puisque l’inclusion entre boules est équivalente à une inclusion entre dômes, nous obtenons la propriété suivante illustrée sur la figure 7.14 [CM07] :

LEMME 7.1 *Une boule B est maximale dans Y si et seulement si son dôme \hat{B} est maximal dans \hat{Y} .*

Dans le cas particulier où Y est une union finie de boules ouvertes, nous réexprimons la condition de maximalité d’un dôme en termes de contact de ce dernier avec la frontière de \hat{Y} . Pour cela, remarquons que, au-dessus du plan $z = 0$, la frontière de \hat{Y} coïncide avec l’enveloppe supérieure des paraboloïdes associés aux boules B contenues dans Y . Cette enveloppe a pour équation $z = h(x, y)$ avec :

$$h(x, y) = \max_{B \subseteq Y} h_B(x, y).$$

Une boule est maximale si et seulement si son dôme touche l’enveloppe supérieure des paraboloïdes, formellement :

LEMME 7.2 (Caractérisation par paraboloïdes [CMC07]) *Soit $Y \subseteq \mathbb{R}^2$ une union finie de boules ouvertes. Soit B , une boule ouverte de \mathbb{R}^2 . Alors :*

$$B \text{ maximale dans } Y \iff \exists (x, y) \in B, h_B(x, y) = h(x, y).$$

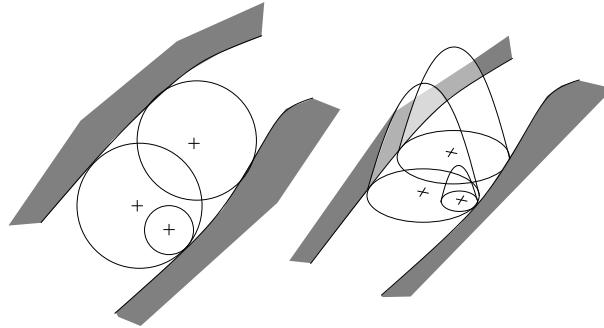


FIGURE 7.14 – Boules incluses dans une forme et dômes associés.

Maintenant, nous allons nous intéresser à une forme discrète $X \subseteq \mathbb{Z}^2$ et notons $Y \subseteq \mathbb{R}^2$ la forme continue obtenue en remplaçant les boules strictes euclidiennes contenues dans X par des boules ouvertes de même centre et de même rayon (voir figure 7.15). Notons $B(P)$ la plus grande boule ouverte centrée en P et contenue dans Y . D'après le lemme précédent, l'ensemble suivant :

$$\text{AM}_0(Y) = \{P \in \mathbb{Z}^2 \mid \exists (x, y) \in X, h_{B(P)}(x, y) = h(x, y)\},$$

forme un sous-ensemble de l'axe médian de Y . Notons cependant que ce sous-ensemble n'est pas forcément inclus dans l'axe médian discret de X . En effet, une boule ouverte peut être maximale dans Y sans pour autant que la boule stricte euclidienne de même centre et de même rayon le soit dans X (voir figure 7.15 à gauche). Ceci nous conduit à introduire l'*axe médian discret* de X comme la restriction de l'ensemble précédent aux points de l'axe médian discret de X , $\text{AMD}(X) = \text{AM}_0(Y) \cap \text{AM}_{de}(X)$. Dans [CM07], nous avons montré que $\text{AMD}(X)$ permet de reconstruire la forme sans perte.

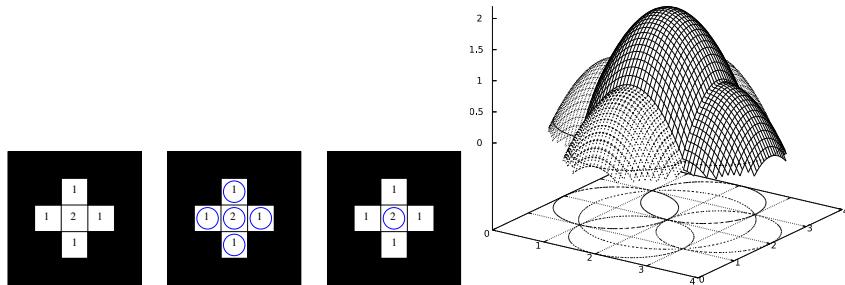


FIGURE 7.15 – De gauche à droite, une forme discrète X et sa transformée en distance euclidienne ; les 5 centres des boules dont le dôme touche l'enveloppe supérieure des paraboloïdes, dessinée à droite ; l'axe médian discret est formé du seul point central encerclé ; forme continue Y et enveloppe supérieure des paraboloïdes bordant la forme \hat{Y} .

Encore une fois, nous allons pouvoir définir des algorithmes séparables pour extraire cet axe médian. L'algorithme procède en trois étapes. Tout d'abord, nous calculons la transformée en distance euclidienne. Après cette étape, nous disposons d'une image $f : [1, m]^2 \rightarrow \mathbb{Z}$ mémorisant, en chaque point discret (x, y) , le carré de sa distance euclidienne au complémentaire de X . Puis, nous calculons l'image $h : [1, m]^2 \rightarrow \mathbb{Z}$ définie par :

$$h(x, y) = \max_{1 \leq i, j \leq m} \{f(i, j) - (x - i)^2 - (y - j)^2\}. \quad (7.3)$$

Avec les notations du paragraphe précédent, ceci revient à calculer, en chaque point discret (x, y) , l'altitude de l'enveloppe supérieure des paraboloïdes associés aux boules B

contenues dans la forme Y . Nous observons une grande similarité entre l'équation (7.2) utilisée pour la reconstruction et l'équation (7.3). Dans les deux cas, l'image h peut se calculer en utilisant l'algorithme séparable en $O(m^2)$ de la section 7.3.1. Néanmoins, nous marquons à présent, pour chaque point (x, y) , les centres P des boules $B(P)$ dont le dôme passe par le point $(x, y, h(x, y))$. A l'issue de cette étape, les points marqués forment l'ensemble $AM_0(Y)$ défini dans la section précédente. Une dernière étape est encore nécessaire pour éliminer de l'ensemble précédent les points ne faisant pas partie de l'axe médian discret, comme illustré figure 7.15.

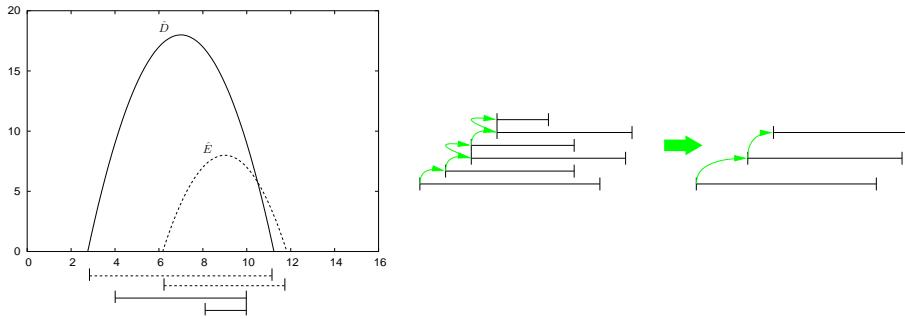


FIGURE 7.16 – Une illustration de la différence entre l'ensemble $AM_0(Y)$ et l'axe médian discret. A gauche : les segments en pointillé représentent les boules euclidiennes 1D associées à $AM_0(Y) = \{\hat{D}, \hat{E}\}$ alors que les segments en traits pleins représentent les boules discrètes 1D. A droite : illustration du processus de filtrage permettant de ne conserver que les boules discrètes 1D maximales.

Pour passer de l'ensemble $AM_0(Y)$ à l'axe médian discret $AMD(X)$, un processus mono-dimensionnel, dimension par dimension, peut-être donné. A l'issue du calcul de $AM_0(Y)$, nous avons un ensemble de disques 1D, c'est-à-dire un ensemble de segments à coordonnées non entières. Sur la figure, si $\{\hat{D}, \hat{E}\}$ forme l'ensemble des dômes contribuant à l'enveloppe supérieure des paraboles, seul le centre de D appartient à l'axe médian discret de la forme 1D. Un processus très simple peut être ajouté à l'algorithme 7.12 pour effectuer le filtrage des segments. La preuve de l'exactitude de l'algorithme précédent est donnée dans [CM07].

A l'issue de ce processus, nous obtenons donc l'ensemble de centres de boules maximales définissant l'axe médian discret, tout en maintenant la reconstruction de la forme.

Les figures 7.17 et 7.18 présentent quelques résultats en dimensions 2 et 3.

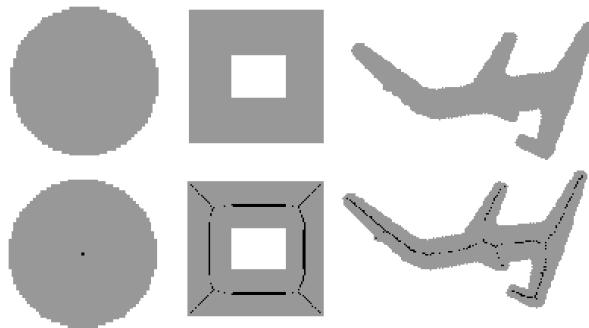


FIGURE 7.17 – Extraction de l'AMD en dimension 2 : la première ligne présente les objets discrets, la seconde présente les centres de boules du AMD (pixels en noir).

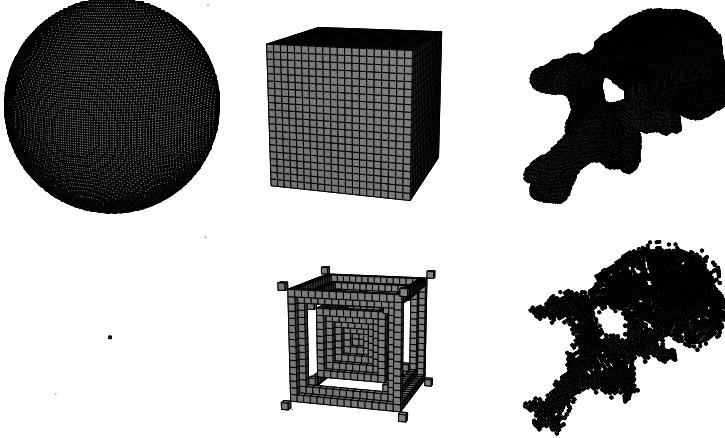


FIGURE 7.18 – AMD en dimension 3 : la première ligne présente les objets discrets, la seconde ligne présente les centres des boules de l’AMD.

7.3.5 Liens avec la géométrie algorithmique

Dans [CM07], nous avons proposé une technique locale de filtrage facilement généralisable en dimension supérieure. Cette technique est en fait présentée pour illustrer les liens entre ces outils de géométrie discrète, et des objets classiques en géométrie algorithmique. En effet si nous faisons le bilan des sections précédentes :

- transformée en distance de $X \Leftrightarrow$ construction d’un diagramme de VORONOÏ de \overline{X} ;
- transformée en distance inverse d’un ensemble $(P_k, r_k) \Leftrightarrow$ lieu de valeurs négatives dans le diagramme de puissance (ou de LAGUERRE) de cet ensemble ;
- extraction de l’axe médian \approx sites des cellules non-vides du diagramme de puissance précédent.

Le premier point ayant été discuté dans la section 7.3.2, revenons sur les deux derniers. Considérons une liste de sites $\{(s_i, r_i)\} \subset \mathbb{R}^n \times \mathbb{R}$ et p un point de l’espace. La *puissance* de s_i sur p est donnée par [Aur87] :

$$\sigma_i(p) = d(p, s_i) - r_i^2. \quad (7.4)$$

Si $\sigma_i(p) < 0$, alors p appartient à la boule ouverte de centre s_i et de rayon r_i . De la même façon que pour la construction du diagramme de VORONOÏ, le *diagramme de puissance* est une partition de l’espace en cellules c_i telles que

$$c_i = \{p \in \mathbb{R}^n : \sigma_i(p) < \sigma_j(p), i \neq j\}. \quad (7.5)$$

Point important, si une boule A est contenue dans une boule B , la cellule dans le diagramme de puissance associée à A sera vide. Dans le cas général, les cellules sont convexes (voir figure 7.19). Cet objet géométrique est un outil très utilisé quand il s’agit de considérer des interactions entre boules [Aur87, BCO⁺96, AE07] ou dans des cas de reconstruction de surfaces [ACK01].

L’équation (7.4) est à mettre en relation avec l’équation (7.1). Ainsi, le calcul de puissance correspond exactement à l’opposé de la reconstruction en distance. Le schéma 7.19-(b) illustre aussi le parallélisme entre la représentation du diagramme de puissance comme l’intersection de paraboloïdes elliptiques opposés à ceux présentés dans la section 7.3.4.

Dans l’énoncé initial, le signe \approx indique que s’il existe des liens très forts dans le cas continu entre les sites n’ayant pas de cellules dans le diagramme de puissance et les boules

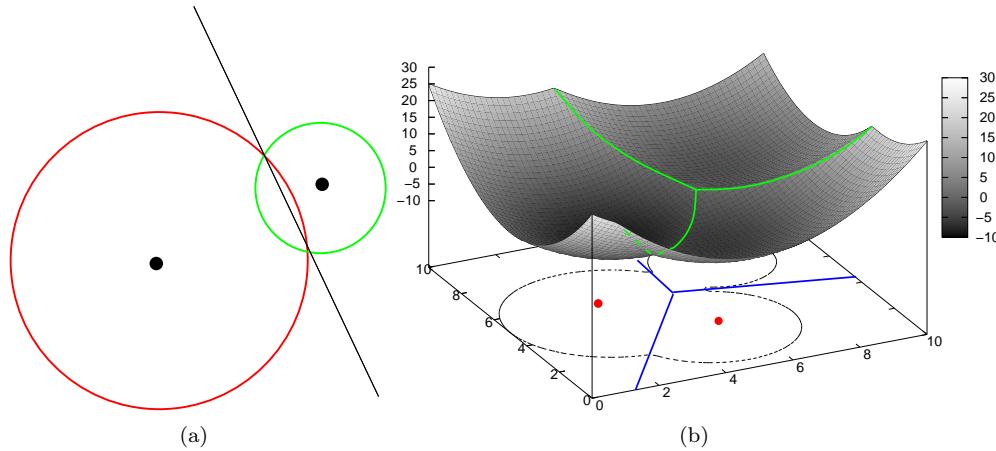


FIGURE 7.19 – (a) Diagramme de puissance associé à deux boules de rayons différents. (b) Représentation surfacique du diagramme avec des paraboloides elliptiques.

de l'axe médian. Le processus de filtrage discret défini dans la section précédente rend cependant l'équivalence moins directe dans le cas discret.

7.4 Généralisation aux espaces toriques

Étant donnée la séparabilité des algorithmes, certaines généralisations sont triviales. C'est le cas par exemple pour le traitement des grilles avec élongations. Il suffit pour cela d'insérer les coefficients d'élongation dans les calculs dimension par dimension.

Dans ce qui suit, nous présentons une généralisation qui tire aussi profit des aspects séparables : généralisation aux espaces toriques à n dimensions. Informellement, le principe est de considérer d'une autre manière les bords des volumes discrets. En effet, dans certaines applications comme la modélisation de matériaux ou de synthèse de textures, on peut s'intéresser à un échantillon particulier qui sera répété par translation régulière sur tout l'espace (voir figure 7.20). L'objectif ici est de faire une analyse volumique (transformée en distance, reconstruction, extraction d'axe médian, ...) sur cet échantillon qui sera ensuite cohérente par translation.

D'un point de vue formel, il nous faut considérer, pour l'échantillon, une grille torique représentée dans la figure 7.20-(c). Ainsi, si nous calculons une transformée en distance par exemple, les informations sur le bord gauche se propageant sur le bord droit, nous aurons un calcul exact dans le cas de la répétition régulière du motif. Notons que ce principe est valable en dimension supérieure.

Grâce à la séparabilité des algorithmes, nous pouvons généraliser tous les processus précédents sur ces grilles grâce à deux principes élémentaires illustrés dans le cas 2D :

- gestion indépendante des dimensions : nous pouvons traiter indépendamment les lignes et les colonnes ;
- dans une même dimension, gestion indépendante des lignes : dans une première phase selon l'axe des x , la *circularité* de chaque ligne peut être traitée de manière indépendante.

Illustré par la figure 7.20-(c), lors du calcul de transformée en distance (ou de reconstruction), il nous faut calculer une minimisation (ou une maximisation) cette fois sur un tableau circulaire. Pour illustrer cela, la figure 7.20-(d) présente un unique disque discret sur un modèle torique.

Nous nous intéressons, pour le problème considéré, à des points particuliers que nous appelons des *points de rupture*. Un point de rupture est un point qui *interdit* la propagation

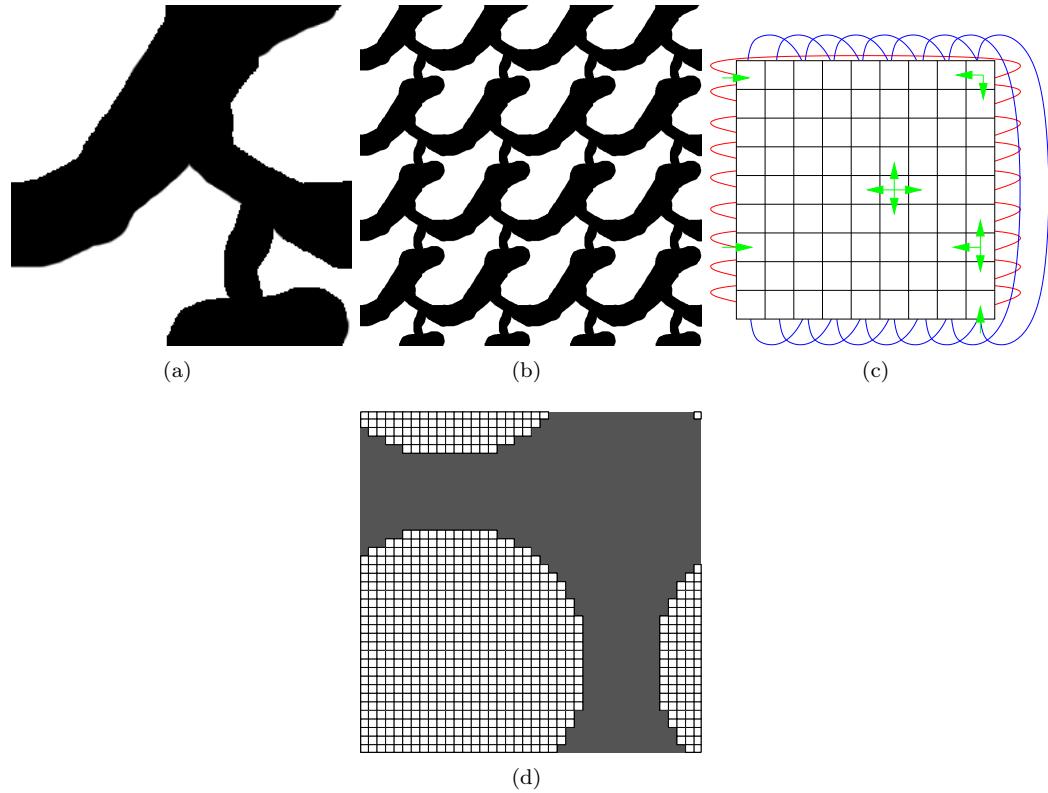


FIGURE 7.20 – (a) et (b) construction d'un pavage de l'espace par la répétition régulière d'un motif (a), (c) représentation de l'espace torique et (d) exemple de disque “torique”.

d'un calcul entre les pixels à sa gauche des pixels à sa droite.

Pour la première étape de la transformation en distance, un point p du complémentaire \bar{X} est un point de rupture. En effet, p peut influer sur sa partie droite et gauche mais aucun pixel à sa gauche n'a d'influence sur les pixels à sa droite. Ainsi, pour calculer sur cet espace torique la transformée en distance monodimensionnelle, il suffit de trouver un point de rupture (point de \bar{X}). Ensuite, nous pouvons réécrire à partir de ce point la ligne (torique) en un vecteur classique auquel on ajoute un point supplémentaire dans \bar{X} (voir figure 7.21). Enfin, nous pouvons calculer la transformation en distance classique sur ce vecteur. Par la nature des points de rupture, nous n'avons pas de propagation d'information entre les extrémités. Finalement, nous pouvons réécrire les valeurs dans la ligne initiale (figure 7.21). Le dernier point de rupture que nous ajoutons nous permet de traiter la propagation d'information de ce point à sa gauche.

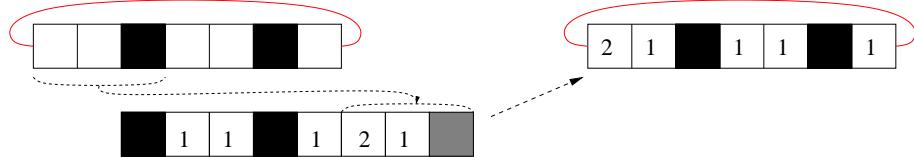


FIGURE 7.21 – Illustration de la première étape pour la transformée en distance torique : sélection du point de rupture, recodage en vecteur et réécriture du résultat.

Pour obtenir globalement le résultat de la première étape de la transformée en distance (algorithme 2), nous traitons les lignes séparément : pour chaque ligne, nous cherchons un point de rupture, nous recodons le vecteur, nous appliquons l'algorithme classique et nous

réécrivons les résultats. S'il n'y a pas de point de rupture (pas de point de \bar{X} dans cette ligne), la transformée en distance n'a pas de valeur sur cette ligne et nous passons à la suivante.

En ce qui concerne l'implémentation, il n'est pas nécessaire de passer par une étape de recopie de ligne : en effet, il suffit de parcourir la ligne pour trouver le premier point de rupture et ensuite nous faisons des parcours et des propagations avec des modulus de la taille de la ligne sur les indices.

Pour les étapes suivantes de la transformation en distance ainsi que pour les algorithmes de reconstruction réversible ou d'axe médian, nous devons pouvoir gérer de manière torique le calcul d'une enveloppe inférieure (resp. supérieure) de paraboles. Dans le premier cas, nous définissons le point de rupture comme étant le centre de la parabole de hauteur minimale. En effet, ce point appartient nécessairement à l'enveloppe inférieure. Il partitionne ainsi la liste circulaire en deux parties. Notons que n'importe quel point dont nous sommes sûrs qu'il sera dans l'enveloppe finale est un point de rupture. Le point de hauteur minimale nous donne cette certitude. Pour la reconstruction en distance ou l'extraction d'axe médian, la sélection des points de rupture est tout aussi triviale : il faut considérer le point qui maximise la hauteur.

La figure 7.22 illustre cette recherche de point de rupture et de recodage dans le cas de l'extraction de l'axe médian ou de la transformation inverse.

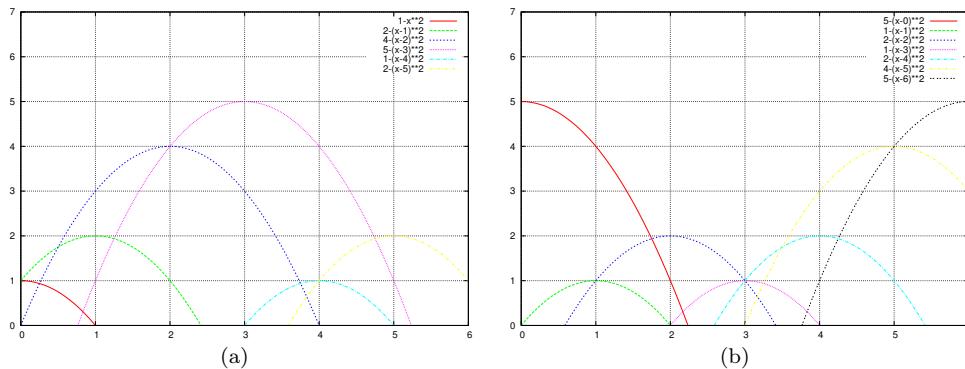


FIGURE 7.22 – Illustration du recodage de la liste circulaire $[1, 2, 4, 5, 1, 2]$ pour le calcul de transformation inverse ou l'extraction de l'axe médian. Le point $i = 3$ est utilisé comme point de rupture.

Notons que l'algorithme global reste identique : pour chaque ligne circulaire, on recherche un point de rupture, on recode (implicitement ou explicitement) un nouveau vecteur de taille augmentée de 1, on applique l'algorithme initial et on réécrit. La figure 7.23 présente un résultat de transformation en distance euclidienne et de l'extraction de l'axe médian sur une grille torique 2D.

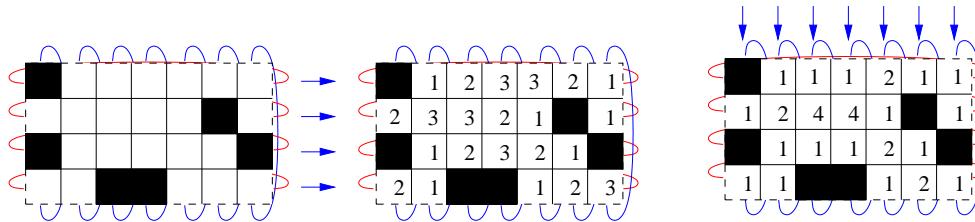


FIGURE 7.23 – Étapes de la transformation en distance sur une grille torique.

En dimension 3, les structures sont plus difficiles à illustrer. Sur la figure 7.25, nous

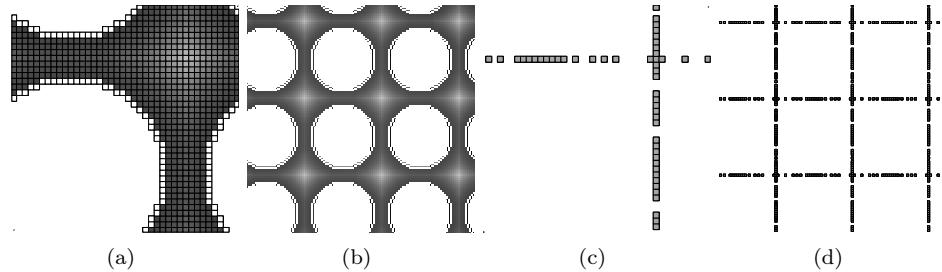


FIGURE 7.24 – Résultat des algorithmes sur le complémentaire du motif de la figure 7.20-(d).
(a) Transformée torique du motif et (b) illustration du pavage de l'espace par translation régulière du résultat. (c) Extraction de l'axe médian torique de l'image (a) et (d) la répétition de l'axe médian.

représentons la transformée en distance d'un motif élémentaire (complémentaire d'un cube de taille 10 et complémentaire de l'objet A1 de taille 50^3) après répétition $3 \times 3 \times 3$ du motif.

La figure 7.26 illustre l'algorithme de transformation en distance inverse. Pour tous ces algorithmes, si l'objet discret initial ne “touche” pas les bords du volume, tous les résultats sont bien évidemment équivalents aux algorithmes classiques. Remarquons aussi que tout ceci est généralisable aux dimensions supérieures.

7.5 Conclusion

Dans ce chapitre, nous avons présenté une série d'algorithmes pour l'analyse volumique d'objets discrets. Le caractère particulier de ces algorithmes est qu'ils sont séparables dimension par dimension. En utilisant cette spécificité, nous avons pu obtenir des complexités optimales et des généralisations aux dimensions supérieures, mais aussi sur d'autres modèles de grille comme les espaces toriques.

Une perspective intéressante serait de construire de manière séparable un diagramme qui associerait, à chaque point discret, le disque de plus grand rayon qui le contient. Intuitivement, cette structure pourrait s'obtenir de manière simple à partir du diagramme de puissance discret. Elle trouve de nombreuses applications en analyse de texture par granulométrie en utilisant la notion de fonction d'ouverture [Soi99].

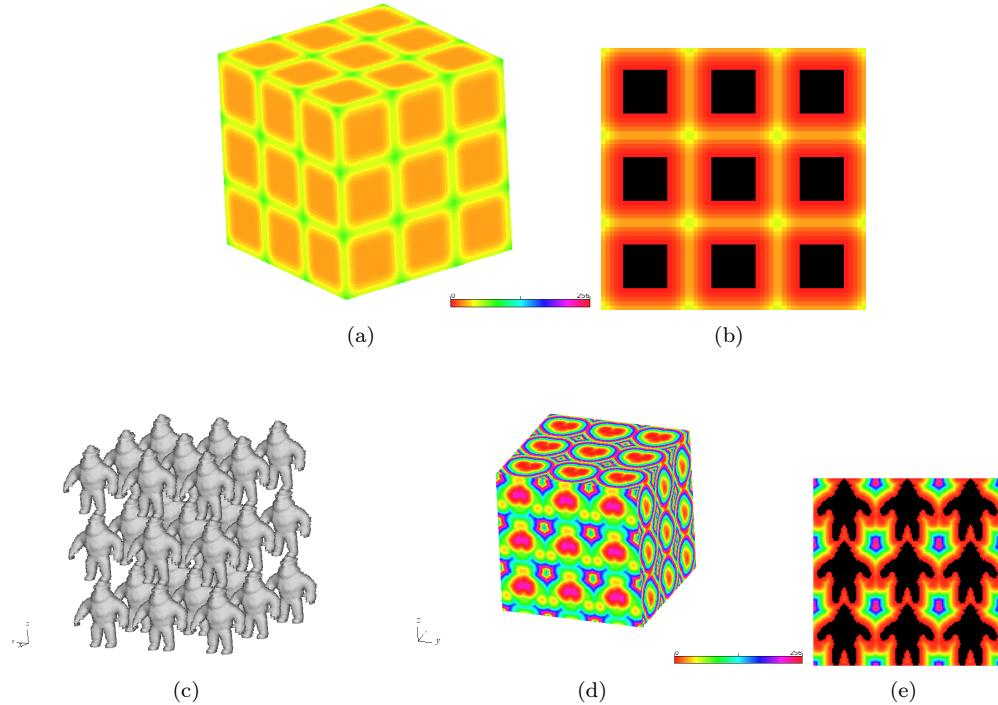


FIGURE 7.25 – Représentation de transformées en distance toriques : (a) transformée sur une répétition d'un motif défini comme étant le complémentaire d'un cube, (b) coupe dans le volume de l'image (a). (c–d) Autre exemple à partir d'un motif défini par le complémentaire de l'objet A1.

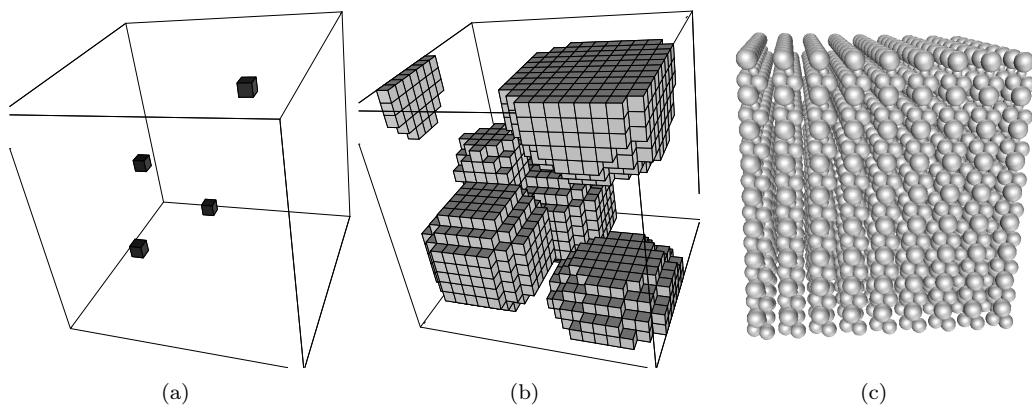


FIGURE 7.26 – Reconstruction sur un espace torique 3D pour générer un texture régulière 3D de sphères : (a) motif avec un placement aléatoire des centres et des valeurs des rayons, (b) transformation inverse et (c) texture obtenue par translation du motif (les sphères discrètes sont représentées par des sphères euclidiennes).

Chapitre 8

Minimalité et simplification de l'axe médian

Sommaire

8.1	Introduction	108
8.2	L'axe médian minimum est NP-complet	108
8.2.1	Schéma de la preuve	109
8.2.2	Définition des variables	109
8.2.3	Définition des liens	110
8.2.4	Définition des clauses	111
8.2.5	Finalisation de la preuve	111
8.3	Simplification réversible de l'axe médian	113
8.4	Filtrage de l'axe médian et liens avec la géométrie algorithmique	114
8.5	Application : compression sans perte et transmission progressive	116
8.5.1	Problématique	116
8.5.2	Primitives à base de boules	117
8.6	Conclusion et perspectives	120

8.1 Introduction

La représentation d'une forme discrète *via* l'union de boules issues de l'axe médian très utilisée dans de nombreuses applications. Cependant, dans un contexte discret où la reconstruction de la forme initiale est exacte dans le sens où tous les points discrets sont couverts par au moins une boule de l'axe médian, celui-ci n'est, bien souvent, pas de cardinalité minimum en nombre de boules. Considérons l'exemple donné dans la figure 8.1 dont les valeurs correspondent à la transformée en distance euclidienne ; les points des valeurs entourées dans la figure de gauche correspondent à l'axe médian discret. Or, dans la figure de droite, le sous-ensemble indiqué permet toujours de reconstruire la forme. Les boules supprimées correspondent à des boules maximales couvertes par une union de boules. L'axe médian discret minimum peut donc être défini comme étant un sous-ensemble de l'axe médian discret possédant un nombre minimum de boules tout en permettant une reconstruction exacte. Notons qu'il n'y a pas unicité de l'axe médian minimum il suffit d'ajouter une colonne à la figure 8.1 pour illustrer cela.

1	1	1	1	1	1	1
1	(4)	(4)	(4)	(4)	(4)	1
1	1	1	1	1	1	1

1	1	1	1	1	1	1
1	(4)	4	(4)	4	(4)	1
1	1	1	1	1	1	1

FIGURE 8.1 – Axe médian discret pour la distance euclidienne (à gauche) et sous-ensemble de l'axe médian permettant une reconstruction exacte (à droite).

Dans la littérature, des travaux ont été menés pour proposer des algorithmes de simplification de l'axe médian [BN95, RB93, BN97, ND97]. Dans ces sections, nous abordons ce problème d'un point de vue théorique, *via* une preuve de NP-complétude pour l'extraction de l'axe médian minimum (section 8.2). Ensuite, nous regarderons les solutions heuristiques pour approcher cet axe minimum. Finalement, la section 8.5 présente quelques applications pour lesquelles nous avons utilisé ces outils.

8.2 L'axe médian minimum est NP-complet

D'un point de vue général, il s'agit de couvrir de manière optimale une forme par des instances d'une famille de formes particulières (ici des boules). En géométrie classique, quand il s'agit de couvrir de manière minimale en cardinalité un polygone avec par exemple des formes convexes, les problèmes sont très souvent NP-difficiles [GO97]. Parmi tous ces problèmes, nous pouvons nous rapprocher des travaux de AUPPERLE ET AL. [ACKO88] concernant la couverture minimale d'un polygone orthogonal (*i.e.* isothétique) par des carrés. Pour faire le bilan de ces résultats, nous avons :

- si le polygone orthogonal est sans trou, le problème peut être modélisé comme un calcul de cliques maximales dans un graphe *chordal*. Ainsi, le problème est polynomial [ACKO88] et est même linéaire au nombre de segments du polygone [BYBH96] ;
- dans le cas général, le problème est NP-complet.

Dans le cas discret, nous pourrions envisager de rapprocher cela du problème de l'axe médian minimum pour la métrique d_∞ . Néanmoins, les résultats d'AUPPERLE ne peuvent être utilisés directement car nous avons à considérer des carrés de taille impaires (centrés sur les points discrets). Cet aspect peut sembler anecdotique mais il est en fait crucial car si on reprenait la modélisation sous forme d'un problème de graphe dans le cas des objets sans trous, nous pourrions trouver des contre-exemples tels que le soit ne serait pas *chordal*.

Dans ce qui suit, nous allons considérer le problème de l'axe médian minimum, que nous pouvons formaliser par :

DÉFINITION 8.1 (Problème de décision k -MA) *Etant donné un objet discret $X \in \mathbb{Z}^n$*

et un entier k , est-il possible de couvrir X par un ensemble de k boules discrètes pour la métrique d considérée ?

Ce problème appartient à la classe NP : en temps linéaire, nous pouvons vérifier si l'ensemble de k boules couvre ou non toute la forme. Nous proposons une réduction polynomiale de toutes les instances de PLANAR-4-3-SAT (voir section 3.5) afin de prouver que le problème de décision ci-dessus est NP-complet. Pour cela, nous allons nous restreindre à la dimension 2 et à la distance euclidienne d_2 .

8.2.1 Schéma de la preuve

Comme indiqué dans la section 3.5, nous allons montrer une réduction polynomiale d'un problème NP-complet connu vers une famille d'instance de k -MA.

L'objectif est ici de construire, pour toute instance ϕ de PLANAR-4-3-SAT, une figure géométrique telle que l'extraction de l'axe médian minimum serait équivalente à résoudre l'instance de PLANAR-4-3-SAT (instanciation des variables pour valider l'expression booléenne). Nous allons donc présenter un schéma géométrique pour les variables de ϕ , un schéma pour les clauses de ϕ ainsi qu'un mécanisme de "lien" pour le plongement géométrique des arêtes. Cette construction géométrique devra avoir une taille polynomiale en la taille de l'instance ϕ . La figure 8.2 illustre ces différentes transformations.

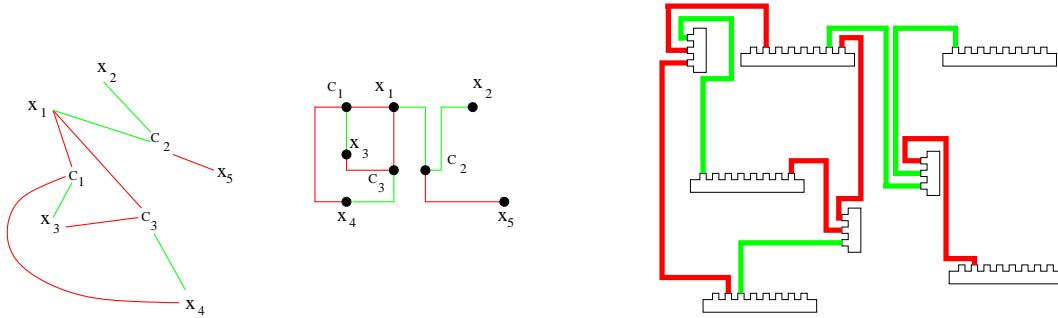


FIGURE 8.2 – Schéma général de la réduction : graphe planaire instance de PLANAR-4-3-SAT, plongement orthogonal [Tam87] du graphe planaire et construction de l'objet discret associé à l'instance.

Cette preuve est issue d'un travail avec JÉRÔME HULIN (LIF, Marseille) et ISABELLE SIVIGNON (LIRIS, Lyon) [CHS07].

8.2.2 Définition des variables

Considérons la figure 8.3 (schéma du haut), l'ensemble de points discrets sous la droite horizontale définit un objet discret que nous appellerons *variable* dans ce qui suit. Sur cet objet, nous pouvons observer 8 tubes verticaux que nous numérotions ces tubes de gauche à droite.

Pour cet objet discret, certains disques maximaux sont obligatoirement dans l'axe médian minimum, ce sont ceux tracés en noir. Pour le reste de la forme, deux couvertures minimales particulières existent (schéma du haut et schéma du bas) : la première va dépasser au niveau de la droite en pointillés sur les tubes impairs et ne pas dépasser les tubes pairs, la seconde couverture est exactement symétrique. Quelle que soit la couverture choisie, nous avons toujours une cardinalité de 72 disques. Ces deux décompositions possibles nous permettent de coder les deux états (Vrai/Faux) d'une variable booléenne. Par convention, le schéma du haut correspond à l'instanciation à Vrai de la variable.

Dans l'expression ϕ , instance de PLANAR-4-3-SAT, les variables apparaissent dans l'expression de manière positive (P) ou négative (N), au plus 4 fois. Par convention, les tubes impairs sont utilisés pour les usages directs de la variable, et les tubes pairs pour les usages de la négation de la variable. Sur la figure 8.3, nous avons 8 tubes nous permettant de couvrir tous les usages de la variable (par exemple P-N-P-N, P-P-N-N, ...) sans avoir à modifier (c'est-à-dire réordonner) la géométrie du plongement des arêtes.

Pour préparer la preuve, remarquons que les tubes sont centrés sur des ordonnées constantes modulo 6 (segments verts). Remarquons aussi que le schéma et les couvertures sont identiques si la figure est tournée de 90° .

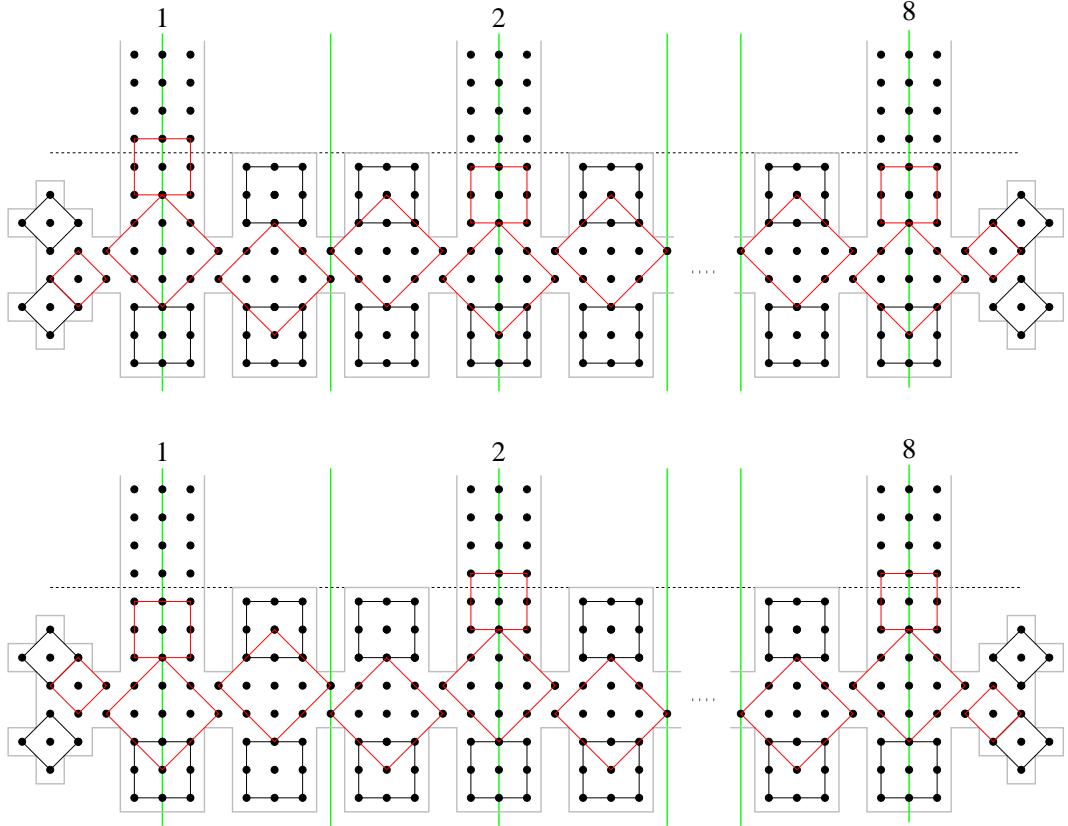


FIGURE 8.3 – Schéma géométrique et couvertures minimales d'un littéral. Le schéma du haut correspond à l'assignation à Vrai de la variable.

8.2.3 Définition des liens

Le décalage que nous observons sur les tubes peut être assimilé à des signaux transmis des variables vers les clauses. Lors du plongement géométrique du graphe planaire induit par ϕ , il nous faut une construction géométrique pour les arêtes permettant de transmettre ces signaux sans les modifier.

La figure 8.4 présente différents schémas permettant de construire ces liens. Nous avons besoin tout d'abord de construire des parties parallèles aux axes. Pour cela, nous utilisons simplement un tube de largeur 3 et de longueur $3L$ (pour $L \in \mathbb{Z}$). Pour cette construction, les couvertures minimales sont de L boules, indépendamment du fait qu'il y ait un décalage ou non. De plus, il existe une couverture minimale telle que les décalages soient préservés d'une extrémité à l'autre. Ensuite, il nous faut pouvoir tourner de 90° ou encore se décaler en préservant l'alignement modulo 6. Tous ces éléments peuvent être combinés pour former un

lien complexe que nous pouvons assimiler à un chemin (1)-connexe sur une grille $6N \times 6N$. Nous reviendrons sur ce point plus tard.

La figure 8.4 illustre ces différents objets. Ce qu'il faut en retirer pour la preuve :

- il n'y a que deux façons de couvrir un lien permettant de transmettre un signal d'une extrémité à l'autre ;
- la cardinalité de la couverture minimale ne dépend pas de la nature du signal (uniquement de la longueur de ces formes) ;
- tous les schémas sont invariants par rotation (90°) ;
- les extrémités des liens sont centrées sur des abscisses et des ordonnées constantes modulo 6.

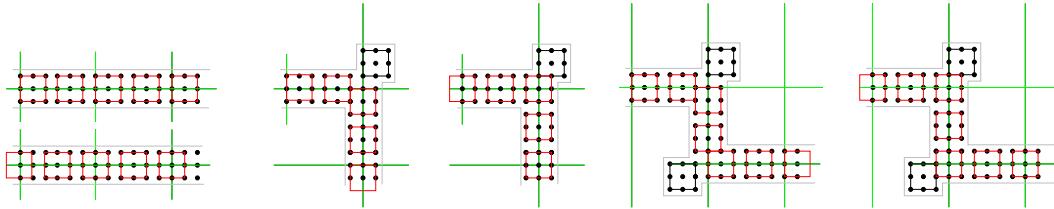


FIGURE 8.4 – Éléments de construction pour définir des liens.

8.2.4 Définition des clauses

Le dernier objet nous permet de définir les clauses dans notre schéma géométrique. L'objectif de cet objet est de prendre en entrée trois liens (les trois littéraux intervenant dans la clause) et de coder intuitivement un “OU” entre les signaux envoyés par les variables.

La figure 8.5 nous permet de faire cela (points discrets à droite de la droite verticale) : trois liens sont connectés sur les parties gauches (toujours centrés modulo 6). Si les trois signaux sont à faux (c'est-à-dire aucun décalage au niveau du trait vertical), la couverture minimale en cardinalité nécessitera 10 boules pour couvrir la forme (9 carrés 3×3 et une boule de rayon $\sqrt{5}$). Si maintenant un des lien transmet un signal à vrai (décalage), il ne nous faut plus que 9 boules pour couvrir la forme. Tous les cas ne sont pas détaillés dans la figure 8.5 mais ils ne portent aucune difficulté. En résumé :

- si un des trois liens transmet un signal vrai, la couverture minium de la clause contient 12 boules ;
- si tous les signaux sont faux, 13 boules seront nécessaires ;
- les liens en entrée sont bien centrés sur un modulo 6 constant ;
- la clause et ses décompositions sont invariantes en rotation (90°).

8.2.5 Finalisation de la preuve

Nous ne détaillerons pas la preuve donnée dans [CHS07] mais nous donnons l'idée générale. Pour tout instance ϕ de PLANAR-4-3-SAT, nous commençons par construire son plongement géométrique orthogonal sur une grille discrète de taille $6lN \times 6lN$ ($l, N \in \mathbb{Z}$). Ce plongement d'un graphe planaire, ayant des sommets de degré inférieur à 4, place les sommets du graphe sur des points discrets et construit des arêtes orthogonales (isothétiques) [Tam87] (voir figure 8.2). Pour chaque élément de ce plongement (sommets et arêtes), nous allons construire un objet discret composé des objets *variables*, *clauses* et *liens*.

Sur ce plongement, nous allons pouvoir associer à chaque arête un chemin (1)-connexe, ces chemins ayant la propriété de ne pas s'intersecter. Ainsi, si $l \geq 1$, les liens de la figure 8.4 ne s'intersectent pas non plus.

Concernant les clauses et les variables, étant donné que ceux-ci ont une taille constante, il suffit de prendre un l adapté pour faire en sorte qu'il n'y a pas d'intersection entre d'une part

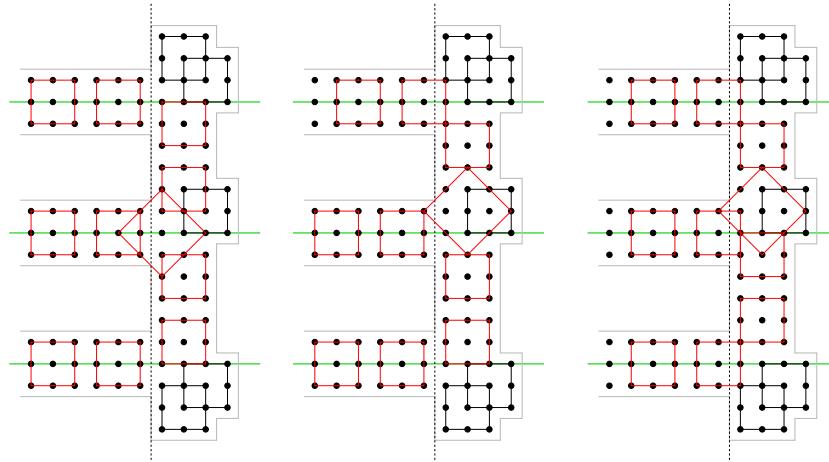


FIGURE 8.5 – Schéma géométrique pour coder les clauses.

ces objets et entre ceux-ci et les liens d'autre part. Cela revient à réduire la résolution de la grille sur laquelle nous avons calculé le plongement géométrique de ϕ . Étant donné que tous les éléments sont construits pour être alignés sur des ordonnées ou des abscisses constantes modulo 6, nous n'aurons aucun problème de *branchement* lors de la construction finale de l'objet. Notons que pour que les liens s'ajustent parfaitement aux variables par exemple (tubes tous du même côté de la variable), il faut localement tordre un peu le plongement orthogonal.

A ce niveau, et en exploitant cette non-intersection entre les objets ainsi que leur invariance par rapport aux rotations de 90° si besoin, nous avons un objet discret (1)-connexe pour tout instance ϕ de PLANAR-4-3-SAT. Cette réduction est polynomiale en temps et en espace car la taille de cet objet est polynomiale en la taille de ϕ .

Pour conclure, il nous faut faire le lien entre une instanciation des variables de ϕ permettant de mettre à vrai la formule, et la couverture minium de l'objet associé à ϕ [CHS07]. Informellement, les variables et les liens ont une couverture ne dépendant pas la valeur de la variable, notons h le nombre de boules nécessaires pour couvrir ces parties de manière minimale. Si une couverture est donnée et en fonction de la convention précisée dans la section 8.2.2, nous allons pouvoir affecter la variable à Vrai ou Faux. Réciproquement et par construction des formes, à partir de toute instanciation des variables permettant de valider un formule ϕ , nous pouvons construire une couverture de l'objet discret associé à ϕ qui soit minimale.

Si nous avons c clauses, nous avons pour chacune d'elles deux couvertures possibles entraînant un nombre total de disques entre $h + c \cdot 9$ et $h + c \cdot 10$. La décomposition atteint le minimum à condition qu'un des littéraux au moins par clause soit à Vrai.

En conclusion, si nous avions un algorithme permettant de construire l'axe médian minimum pour la distance euclidienne, nous pourrions l'appliquer sur l'objet géométrique issu d'une instance quelconque de PLANAR-4-3-SAT et ainsi trouver une instanciation des variables de cette instance pour valider la formule. Ainsi, le problème d'axe médian minimum est au moins aussi difficile que celui de PLANAR-4-3-SAT. Nous pouvons donc conclure la preuve en disant que le problème k -MA minimum en distance euclidienne est NP-complet.

Nous reviendrons dans la conclusion de ce chapitre sur une discussion de cette preuve.

8.3 Simplification réversible de l'axe médian

Bien que le problème soit NP-complet, nous pouvons bien évidemment définir des algorithmes permettant de simplifier l'axe médian tout en maintenant la réversibilité de ce dernier. Ce que nous dit la preuve précédente, c'est qu'un tel algorithme, supposé polynomial en temps, ne produira pas l'axe médian minimum dans le cas général. Dans cette section, nous faisons une analyse des techniques existantes.

Dans [RB93] les auteurs ont proposé une heuristique très rapide qui peut-être décrite très simplement : nous construisons tout d'abord une carte de couverture dans laquelle pour chaque point discret, nous comptons le nombre de boules de l'axe médian contenant ce point. Intuitivement, s'il existe un point P couvert par une seule boule B , B est nécessairement dans l'axe médian minimum (sinon la réversibilité ne serait pas maintenue). L'algorithme procède ensuite de la façon suivante : pour chaque boule B de l'axe médian prise dans l'ordre croissant des rayons, nous vérifions si B n'est pas nécessaire (c'est-à-dire si tous ses points discrets ont une couverture supérieure ou égale à 2). Si B n'est pas nécessaire, nous supprimons ce disque de l'axe médian et nous décrémentons la carte de couverture en ses points discrets de 1. Ce processus est itéré sur l'ensemble des boules. Si l'axe médian est donné par $\{B_i\}_{i=1..m}$, la complexité de cette approche est en $O(m \log m + \sum_{i=1}^m |B_i|)$. D'un point de vue de l'approximation, aucune borne n'existe quant à la qualité de la simplification par rapport à l'axe médian minimum.

Une autre approche peut être envisagée en voyant le problème comme une instance particulière du problème de couverture minimale d'ensembles MINSETCOVER [CLR90] : soit X un ensemble de points et \mathcal{F} une famille de sous-ensembles de X telle que tout élément de X est contenu dans au moins un sous-ensemble de \mathcal{F} . La question est ici de trouver l'ensemble \mathcal{F}^* de cardinalité minimale tel que tous les points restent couverts. Ce problème est bien sûr NP-complet. Notons que d'un point de vue méthodologique, pour prouver la NP-complétude de l'axe médian, il paraîtrait naturel de réduire les instances de MINSETCOVER plus proche de notre problème que PLANAR-4-3-SAT. Or cela n'est pas le cas, il est parfois préférable de réduire des problèmes comme 3-SAT que de tenter de réduire des problèmes “trop proches”.

Donc si le problème MINSETCOVER est tout aussi NP-complet que le nôtre, il existe une heuristique permettant d'approcher l'optimum d'un certain facteur. Ce processus est une approche gloutonne donnée dans l'algorithme 5. Si nous notons $H(d) = \sum_{i=1}^d \frac{1}{i}$ et $H_{\mathcal{F}} = H(\max |S|, S \in \mathcal{F})$, alors nous pouvons prouver que l'algorithme glouton du MINSETCOVER a pour borne $H_{\mathcal{F}}$. En d'autres termes :

$$|\hat{\mathcal{F}}| \leq H_{\mathcal{F}} \cdot |\mathcal{F}^*|$$

Algorithme 5 : Algorithme glouton pour le MINSETCOVER.

Données : X et \mathcal{F}

Résultat : l'ensemble $\hat{\mathcal{F}}$ approximant la solution

$U = X;$

$\hat{\mathcal{F}} = \emptyset;$

tant que $U \neq \emptyset$ faire

choisir $S \in \mathcal{F}$ qui maximise $|S \cap U|$;

$U = U - S;$

$\hat{\mathcal{F}} = \hat{\mathcal{F}} \cup \{S\};$

retourner $\hat{\mathcal{F}}$

Bien que cette borne ne soit pas constante et dépende de l'entrée ($H_{\mathcal{F}} \leq (\ln |X| + 1)$), ce résultat nous permet de construire un axe médian simplifié ayant une qualité quantifiée par rapport à l'optimal. De plus et pour le problème spécifique du MINSETCOVER, un preuve existe précisant que le problème n'est pas approximable en $(1 - \epsilon) \ln |\mathcal{F}|$ (pour tout $\epsilon > 0$) ce qui renforce la puissance de l'algorithme glouton dans l'absolu [Fei98].

D'un point de vue algorithmique, une implémentation de l'algorithme 5 existe en $O(\sum_{i=1}^m |B_i|)$ en mettant en place une structure de tri par dénombrement [CLR90]. Néanmoins, les structures de données sont un peu plus complexes ce qui explique les temps de calcul donné en figure 8.6.

La figure 8.6 présente quelques résultats de simplification avec les deux algorithmes précédents. Si l'algorithme glouton nous semble théoriquement plus “performant” car propageant plus d'information lors de la sélection d'une boule (suppression complète des points de la boule et donc remise en question des tailles des autres boules dans le calcul de $|S \cap U|, \dots$), les résultats montrent que l'algorithme de RAGNEMALM ET BORGEFORS [RB93] donne des résultats sensiblement meilleurs.

L'intérêt de l'algorithme glouton est cependant de montrer qu'une borne théorique d'approximation existe. Il serait intéressant de poursuivre l'analyse de cette heuristique pour encore gagner sur la simplification tout en maintenant cette borne.

Objet	$\mathcal{F} = \text{AMD}(X)$	$\hat{\mathcal{F}}$ RAGNEMALM ET AL.	$\hat{\mathcal{F}}$ glouton

FIGURE 8.6 – Expérimentations de différents algorithmes de simplification d'axe médian : (*de gauche à droite*) les objets discrets, l'axe médian discret, la simplification de [RB93] et l'algorithme glouton. Le nombre de boules et le taux de simplification sont donnés sous les objets. Le temps de calcul est indiqué entre crochets.

8.4 Filtrage de l'axe médian et liens avec la géométrie algorithmique

Si maintenant nous levons la contrainte de réversibilité, nous entrons dans la catégorie du filtrage de l'axe médian. L'idée est toujours de supprimer les boules de l'axe médian mais cette fois, nous pouvons être amenés à perdre de l'information lors de la reconstruction. Sans plus d'hypothèses sur le contexte, nous quantifierons cette perte par la distance de

HAMMING entre les deux objets.

De manière générale, nous pouvons formaliser cela comme un problème d'optimisation : étant donnés un objet X et un seuil λ , trouver un ensemble de boules \mathcal{B} avec $|\mathcal{B}|$ minimum tel que la distance entre X et la transformée en distance inverse de \mathcal{B} est inférieure à λ . Cet énoncé est très générique car nous n'avons pas précisé par exemple que \mathcal{B} devait être un sous-ensemble de l'axe médian de X .

Pour répondre à ce problème avec des algorithmes utilisables en pratique, nous allons faire l'hypothèse que $\mathcal{B} \subset \text{AMD}(X)$ et que la sélection d'une boule de $\text{AMD}(X)$ se fera sur des critères géométriques uniquement. Par ailleurs, nous évaluerons notre résultat avec une distance de HAMMING quantifiant la part de l'objet qui a été “perdue”.

En géométrie algorithmique dans le cas continu, une approche classique consiste à attacher deux mesures à la boule de centre s de l'axe médian (voir figure 8.7) [Att95, AM97] :

- l'étanchéité $\rho(s)$ correspondant au rayon de la boule ;
- l'angle bissecteur $\alpha(s)$ correspondant à l'angle des deux points de contact entre la boule et le bord de la forme.

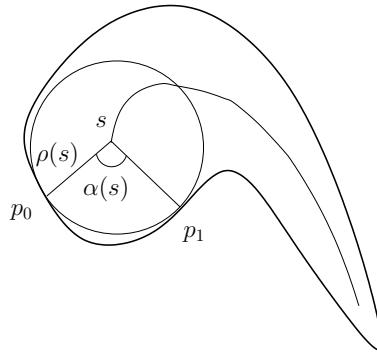


FIGURE 8.7 – Paramètres associés à un point s de l'axe médian : l'étanchéité $\rho(s)$ et l'angle bissecteur $\alpha(s) = \widehat{p_0 s p_1}$.

Dans le cas discret, plusieurs points de contact existent et ils nous sont donnés par la fonction $\Pi_{\overline{X}}$ dans les travaux de la section 7.3.2. Dans [CCZ07] nous avons surtout utilisé ce processus pour guider un amincissement homotopique et non pour effectuer une simplification. Néanmoins, les problématiques sont similaires : dans les deux cas il faut trouver un ordre sur les boules caractérisant leur importance vis-à-vis de la forme.

En considérant les liens avec la géométrie algorithmique présentés dans la section 7.3.5, nous avons proposé dans [CM07] un filtrage simple et facilement généralisable aux dimensions supérieures. Cette technique reprend deux mesures associées à chaque boule : l'étanchéité $\rho(s)$ et l'aire de la cellule associée à s dans le diagramme de puissance restreint à la forme. L'idée sous-jacente est que la géométrie et l'aire de la cellule sont fonction de la position et de la hauteur de la paraboloïde elliptique relativement à ses voisines.

Dans le cas discret, de la même façon que pour l'étiquetage de VORONOÏ, l'algorithme d'extraction de l'axe médian discret nous construit le diagramme de puissance en extension (nous avons un étiquetage de tous les points discrets mais pas la connaissance du graphe complet). Ainsi, nous pouvons très simplement associer à une cellule (*i.e.* à chaque boule) son aire estimée par le nombre de points discrets ayant l'étiquette correspondante (mesure notée $\kappa(s)$).

Ainsi, comme dans [AM97], nous avons un filtrage à deux seuils, l'un sur l'étanchéité (ρ_0), l'autre sur l'aire de la cellule associée à la boule (κ_0). La figure 8.8 donne une illustration en 2D du filtrage avec quelques reconstructions intermédiaires. Le tableau 8.4 et la figure 8.9 donnent des résultats quantitatifs et des filtrages en dimension 3.

La poursuite de ces travaux en utilisant des résultats de géométrie algorithmique comme

par exemple les fonctions d'inclusion/exclusion de boules [AE07], est une perspective intéressante.

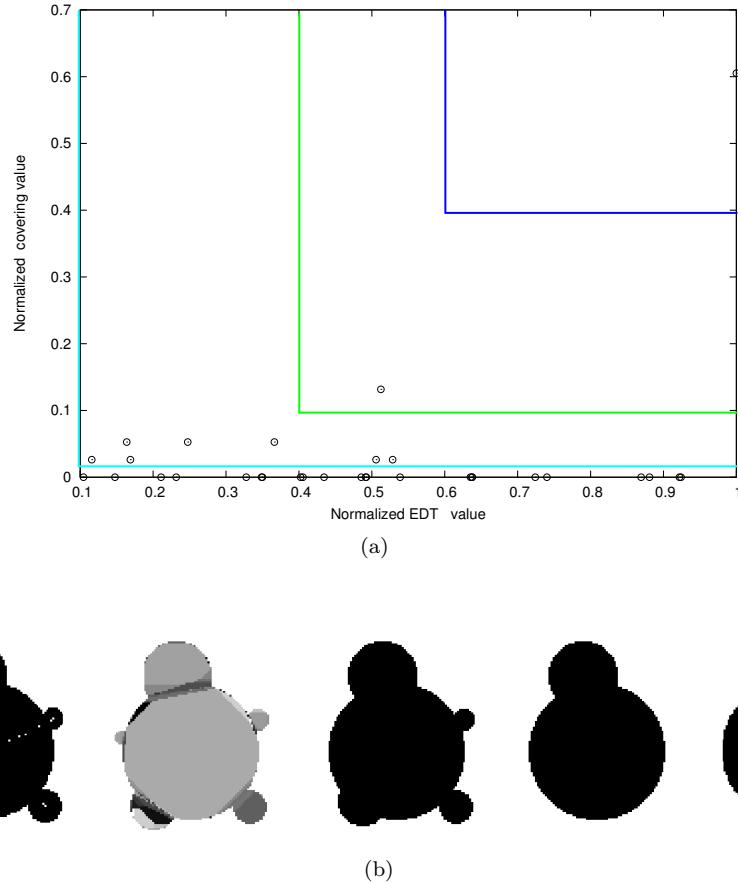


FIGURE 8.8 – Processus de filtrage en dimension 2 : (a) le graphe de filtrage où chaque disque de l'axe médian correspond à un point en fonction de l'étanchéité (axe horizontal) et de l'aire normalisée de la cellule du diagramme de puissance associée. (b), de gauche à droite, la forme discrète originale et son axe médian (38 disques), l'étiquetage par le diagramme de puissance (une couleur par cellule) et trois reconstructions en ne conservant que les disques inclus dans les trois zones rectangulaires de la figure (a) avec respectivement 9, 2 et un disque.

8.5 Application : compression sans perte et transmission progressive

Nous illustrons ici une application à ces représentations d'objets par union de boules dans un contexte de transmission progressive de formes discrètes sur des réseaux bas débits. Ces travaux ont été menés en collaboration avec FLORENT DUPONT lors du stage de Master 2 de LAURENT JOSPIN.

8.5.1 Problématique

Comme nous avons pu le voir dans la section 7.3.5, le filtrage ou la simplification de l'axe médian consiste généralement à quantifier l'importance d'une boule par rapport au

ρ_0	κ_0	nombre de disques		
0	0	38		
0	0.004	9		
0.4	0.1	2		
0.6	0.4	1		
Objet	ρ_0	κ_0	nombre de boules	nombre de voxels
Cube de côté 20	0	0	624	8000
	0.5	0.0005	24	6200
	0.5	0.025	8	5112
Vertèbre	0	0	6593	103302
	0.1	0	5096	100055
	0.2	0.0005	304	78535
	0.5	0.001	75	41873
“Al” 200x200x200	0	0	17098	549162
	0.1	0	8057	514068
	0.1	0.001	197	458541
	0.5	0.001	132	328823

TABLE 8.1 – Résultats quantitatifs des filtrages sur les objets des figures 8.8 et 8.9. Notez que si les seuils sont de $(0, 0)$, le résultat est exactement l’axe médian discret.

reste de la forme et ainsi à ordonner les points de l’axe médian. Dans cette application, nous nous intéressons à la transmission sans perte de volume discret sur un réseau bas débit. Nous ajoutons une contrainte supplémentaire qui est que la transmission se doit d’être progressive. En d’autres termes, nous avons, au niveau du récepteur, une visualisation de la forme en cours de transmission. Ce contexte nous impose que la métrique utilisée pour caractériser l’importance d’une boule est visuelle. Faute de mieux, nous utilisons encore la notion de distance de HAMMING.

L’idée est d’ordonner les boules de l’axe médian et de les transmettre une à une. Dans [DGB03], l’algorithme d’ordonnancement est similaire à l’algorithme 5 : le volume intrinsèque d’une boule à un instant donné est l’apport, en nombre de points discrets de cette boule à la reconstruction totale. A chaque étape, nous sélectionnons et transmettons la boule qui maximise son volume intrinsèque. Ensuite, cette boule est supprimée et les volumes intrinsèques sont mis à jour. Lors de la transmission d’une boule, nous avons un processus de codage permettant de réduire la taille des informations transmises.

8.5.2 Primitives à base de boules

La technique précédente donne de bons résultats mais nous nous sommes intéressés à une approche de plus haut niveau en essayant de reconnaître et de transmettre des interpolations linéaires de boules. Plus précisément, nous allons caractériser la forme par des troncs de cônes 8.10-(a), de triangles 8.10-(b) ou de tétraèdres de boules 8.10-(c). Le tronc de cône par exemple, est défini par deux boules extrémités et par une interpolation linéaire des positions des centres et des rayons. Cette forme est à mettre en relation avec la notion de *gyixel* de THIEL [Thi94]. Dans ce qui suit, nous ne nous intéresserons qu’aux troncs de cône et aux triangles 3D. Les autres formes (triangles 2D et tétraèdres) seront générées dans un second temps.

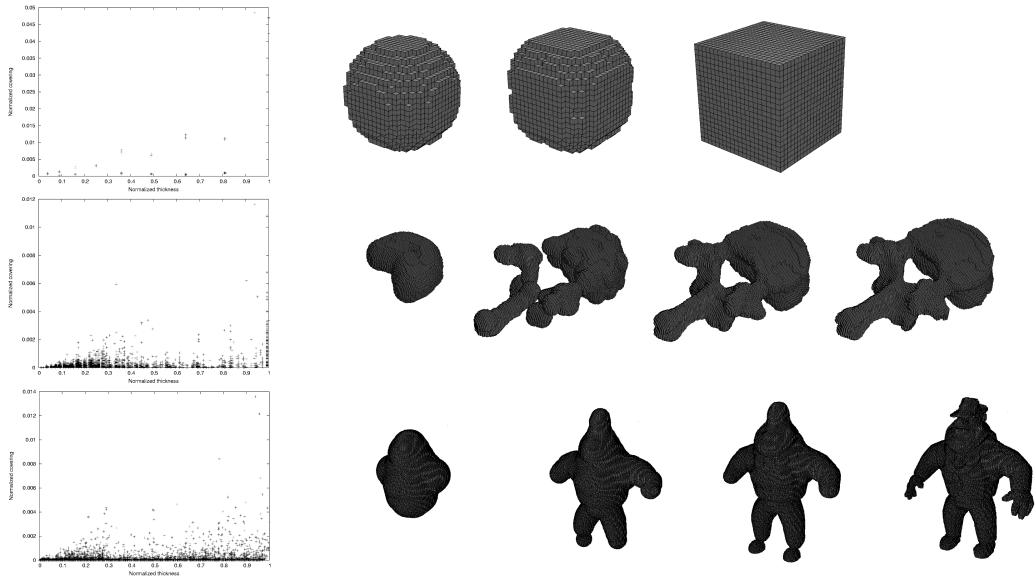
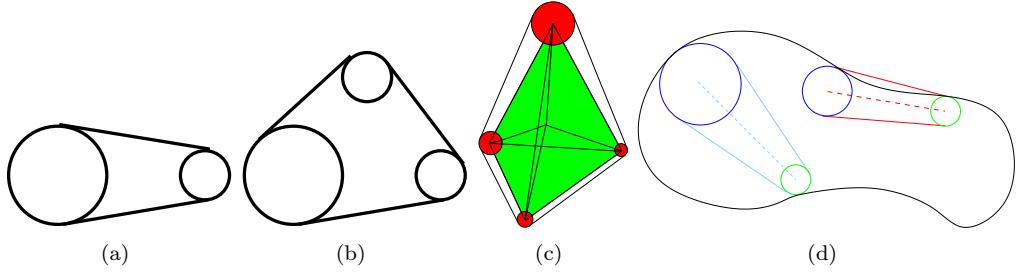


FIGURE 8.9 – Illustration du filtrage en dimension 3 : graphes de filtrage et reconstruction du plus grossier au plus fin des objets discrets.



Dans notre contexte de transmission sans perte, il nous faut une définition de ces objets menant à des algorithmes exacts de reconnaissance. Par reconnaissance, nous entendons que nous devons être capable de décider si une forme géométrique (*e.g.* un tronc de cône) est incluse ou non dans la forme discrète. Si c'est le cas, nous transmettons cette primitive et itérons le processus (modulo un ordonnancement).

Bien évidemment, les boules supports de ces formes seront issues de l'axe médian discret simplifié. Après plusieurs tentatives, il s'est avéré qu'une reconnaissance exacte, par exemple de la discréttisation d'un tronc de cône euclidien (basé sur des boules de l'axe médian discret), n'était pas possible : nous pouvions définir des algorithmes complexes nécessaires mais non suffisants. Nous avons donc opté pour une redéfinition discrète de ces formes en considérant des pseudo-troncs de cônes, des pseudo-triangles et des pseudo-tétraèdres.

Le pseudo-tronc de cône en dimension 3, est défini comme étant formé du segment de droite discrète 3D (0)-connexe [Deb95]. A chaque point discret, nous associons un rayon de boule de manière à ce que l'ensemble des points discrets obtenus au final soient inclus dans le tronc de cône euclidien exact. Grâce à cette inclusion, si un tronc de cône euclidien (à base dans l'axe médian discret) est inclus dans la forme discrète, alors son pseudo-tronc de cône l'est aussi. Pour la reconnaissance d'un tel objet, il nous suffit de parcourir le segment discret 3D (défini de manière canonique) dans la transformée en distance et de vérifier que

toutes les valeurs de distance trouvées sont supérieures aux valeurs construites par notre interpolation

Pour la définition d'un pseudo-triangle 3D, nous procérons de la même manière : nous construisons un triangle discret (voir par exemple [BBN00] ou [And00]) et nous affectons à chaque voxel un rayon tel que l'union convexe euclidienne contienne la forme discrète (voir figure 8.11).

Concernant le récepteur, ce dernier va recevoir par exemple les extrémités d'un pseudo-tronc de cône, il utilise le processus de construction des pseudo-troncs de cône pour tracer le segment discret 3D avec ses valeurs et enfin l'algorithme de reconstruction (section 7.3.3).

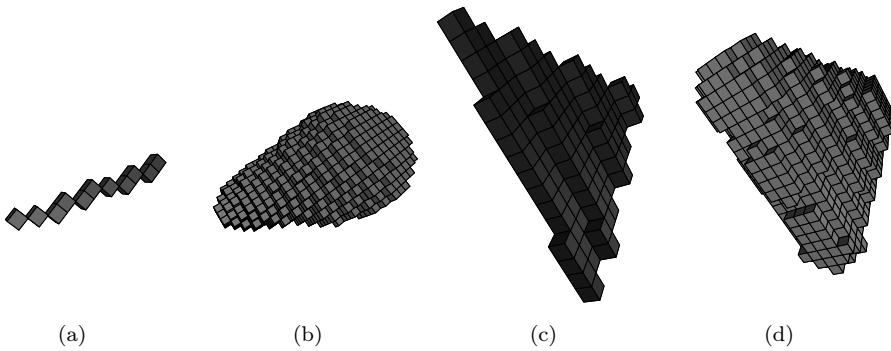


FIGURE 8.11 – Pseudo-tronc de cône et pseudo-triangle 3D.

Pour définir des formes plus complexes comme des unions convexes de trois boules dans le plan ou de quatre boules en 3D, nous exploitons les pseudo-troncs de cône et les pseudo-triangles 3D (voir figure 8.12) : pour reconnaître un pseudo-triangle 2D, nous allons reconnaître les trois pseudo-troncs de cône définis par les trois extrémités, et par le centre du cercle inscrit au triangle tangent aux boules. Dans le cas continu, une telle reconnaissance est exacte ; dans le cas discret, l'objet discret ainsi défini aura la propriété d'être inclus dans la forme continue considérée. Pour la reconnaissance de tétraèdres, nous allons utiliser le même mécanisme : pour reconnaître (et ainsi définir) un pseudo-tétraèdre 3D, nous allons reconnaître les 6 pseudo-triangles 3D formés par les "arêtes" du tétraèdre et la sphère incluse au tétraèdre tangent. Bien sûr, ces constructions n'ont de sens que pour des cas non dégénérés : l'enveloppe convexe des trois disques a 3 parties linéaires dans le cas d'un triangle 2D et l'enveloppe convexe des 4 boules a bien 4 faces pour le tétraèdre.

Nous avons maintenant tout un ensemble de formes, basées sur des combinaisons convexes de boules incluses, à notre disposition pour proposer des codages efficaces de formes discrètes. D'un point de vue théorique, si les boules à l'origine de ces formes sont maximales, il nous faut poursuivre l'analyse pour obtenir des propriétés de maximalité pour les troncs de cône. Typiquement, dans le continu, si les boules sont maximales, les troncs de cônes inclus sont trivialement maximaux. Dans le cas discret, l'implémentation est moins triviale.

Revenons à notre objectif de description de formes. Nous avons des primitives et il nous faut maintenant une heuristique de segmentation minimale de l'objet discret en de telles primitives. Les solutions à ce problème sont extrêmement heuristiques car les choix sont nombreux (Quelle forme ? Quelle stratégie ? ...). Le reste de l'algorithme de transmission reste quant à lui similaire : nous allons transmettre les primitives de volume intrinsèque maximal.

Le problème est maintenant combinatoire : en dimension 3, nous ne pouvons pas explorer toutes les combinaisons des boules entre elles. Pour cela, plusieurs stratégies ont été proposées dans [Jos06] mais nous ne les détaillerons pas. A l'issu d'un codage naïf des primitives, la figure 8.13 illustre simplement l'intérêt de leur construction de telles primitives pour la transmission avec le tracé de courbes débit/distorsion : l'abscisse correspond à la quantité

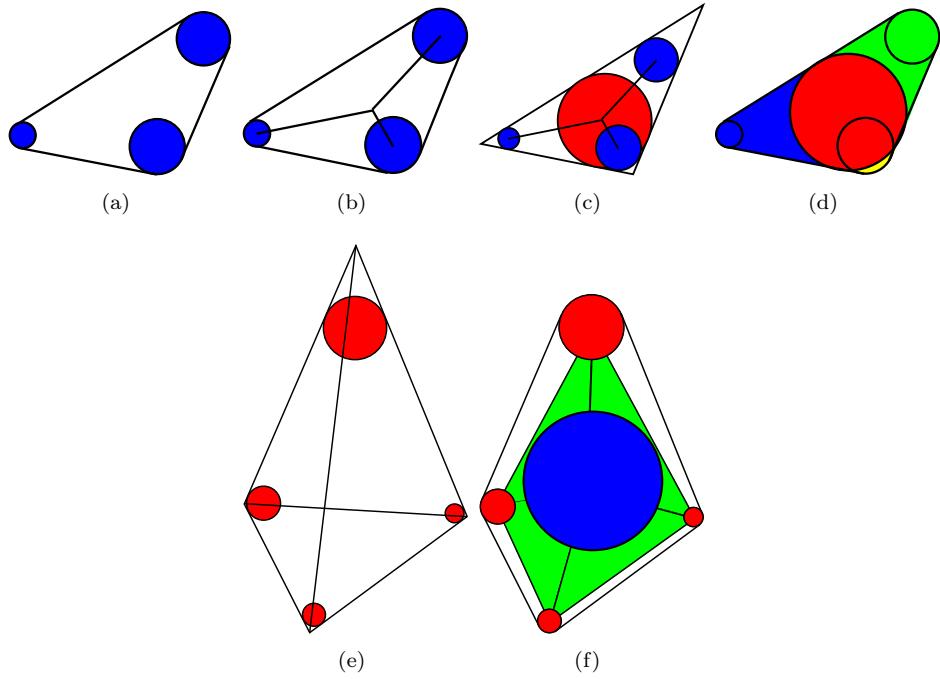


FIGURE 8.12 – Construction d'un pseudo-triangle 2D et d'un pseudo-tétraèdre 3D en utilisant les pseudo-troncs de cônes et les pseudo-triangles 3D.

d'information transmise et l'ordonnée à la distance normalisée entre l'objet reconstruit et l'objet initial. Sur ces graphes, nous voyons l'intérêt d'utiliser les primitives supérieures (troncs, triangles 3D, tétraèdres) par rapport à la transmissions des boules. Néanmoins, sur le second objet, on peut observer que le gain pour les unions convexes de 4 boules par exemple n'est pas significatif par rapport aux troncs de cône. La figure 8.14 illustre la reconstruction progressive de l'objet 32dodge avec les primitives de haut niveau.

L'objectif était initialement de poursuivre cette extraction de primitives à des unions convexes de n boules mais malheureusement, les gains montrés sur les graphes 8.13 ne sont pas pour l'instant suffisant par rapport au coût algorithmique de la recherche combinatoire.

8.6 Conclusion et perspectives

Dans ce chapitre, nous avons présenté des algorithmes permettant de décrire de manière volumique, des objets discrets. Pour simplifier le propos, après quelques variations autour du thème des algorithmes séparables et de leurs généralisations, nous nous sommes attachés à la description de formes par union de boules.

Quel que soit l'objectif, simplification, filtrage, transmission progressive, et que ce soit d'un point de vue théorique ou pratique, l'aspect crucial était de caractériser l'importance d'une boule relativement aux autres et à la forme. Pour mieux comprendre les différentes interactions entre boules, les liens que nous avons présentés avec les diagrammes de puissances nous semblent très importants. Une analyse un peu plus fine de cette structure en interaction avec le modèle discret est une perspective importante. En particulier, il serait intéressant d'analyser les variantes discrètes des formules d'inclusion/exclusion de ATTALI ET EDELSBRUNNER [AE07] pour mieux contrôler l'apport d'une boule sur le volume total de l'objet.

Une autre utilisation de ces objets de la géométrie algorithmique concerne la description

hiérarchique par union de boules d'une forme (*sphere-tree*). Des travaux récents dans ce domaine ont montré l'intérêt d'une telle structure dans de nombreuses applications [BO02]. Une perspective de tous ces outils serait donc de proposer une approche discrète de cette structure.

D'un point de vue plus théorique, nous avons prouvé que, dans le cas général, l'extraCTION de l'axe médian minimum était un problème NP-complet pour la distance euclidienne. Si des généralisations à d'autres distances (d_∞ , distances de chanfrein,...) peuvent être envisagées, elles demandent très souvent une construction *ad-hoc* des objets élémentaires (variables, clauses,..., voir [CHS07]). Par ailleurs, la réduction polynomiale construit des formes possédant des trous (cycles dans PLANAR-4-3-SAT), nous pouvons donc nous poser la question suivante : le problème est-il toujours NP-complet si l'objet n'a pas de trous ? Dans le cas continu, certains problèmes semblent plus simples pour ce type d'objets. Dans le cas discret, si certains signaux ne sont pas très optimistes (non *chordalité* du graphe), le problème reste à étudier.

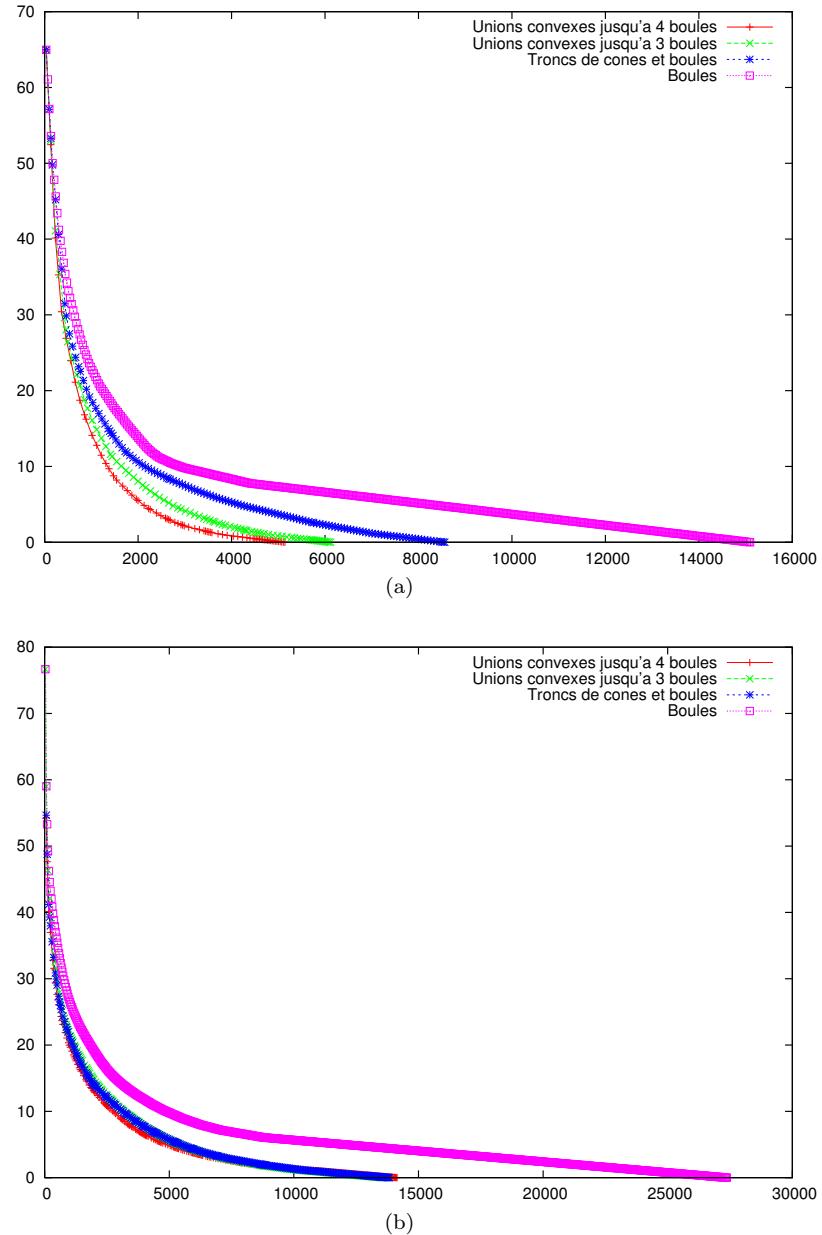


FIGURE 8.13 – Courbes débit/distorsion sur les objets **catenoid** de taille 8 (a) et 32Dodge (b) avec plusieurs stratégies.

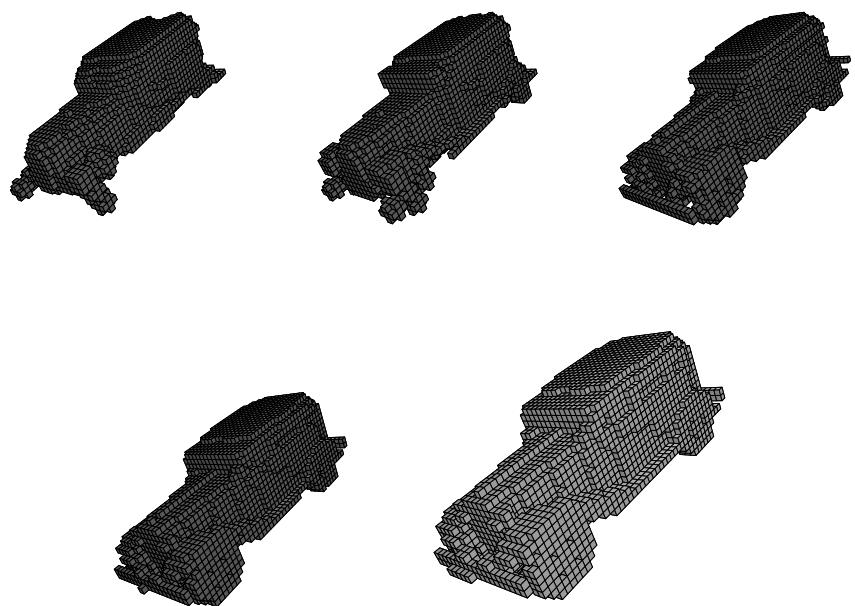


FIGURE 8.14 – Reconstitution de l’objet `dodge` après 1kb, 2kb, 5kb, 10kb, et complète.

Conclusion générale et perspectives

8.7 Synthèse des travaux réalisés

Commençons par reprendre les différents chapitres afin d'isoler un peu plus précisément nos différentes contributions.

- CHAPITRE 4 : nous avons présenté les différents modèles de discréétisation les plus utilisés. Nous avons ensuite proposé une approche basée sur l'arithmétique d'intervalles pour caractériser la supercouverture d'un objet euclidien.
- CHAPITRE 5 : nous nous sommes intéressés aux objets fondamentaux de la géométrie discrète que sont les droites, les plans et les cercles discrets. Après avoir fait le lien avec des résultats de géométrie algorithmique, de programmation linéaire ou d'arithmétique, nous avons analysé la complexité des algorithmes de reconnaissance. Plus précisément, nous avons insisté sur la caractérisation de la préimage des plans discrets et sur l'algorithmique associée à la reconnaissance de cercles discrets.
- CHAPITRE 6 : nous avons mis en œuvre les algorithmes de reconnaissance dans le cadre de la reconstruction réversible d'objets euclidiens. Dans un contexte de tracé par un solveur d'arithmétique d'intervalles de fonctions complexes, nous avons présenté un mécanisme de reconstruction sur des grilles isothétiques irrégulières en dimension 2 à la fois efficace en nombres de segments mais aussi correct topologiquement et géométriquement. Nous avons ensuite exploité et optimisé la préimage des plans discrets pour obtenir une reconstruction réversible d'un polyèdre à partir d'un objet discret 3D.
- CHAPITRE 7 : nous nous sommes intéressés à l'analyse volumique de formes discrètes par le biais de la transformation en distance et de l'extraction de l'axe médian. Nous avons présenté des algorithmes séparables, optimaux en temps, pour tous ces problèmes. Par la suite et en exploitant le côté séparable des outils, nous avons généralisé ces transformations aux espaces toriques, modélisation très utilisée dans de nombreuses applications.
- CHAPITRE 8 : nous nous sommes intéressé à l'exploitation et à l'optimisation de la représentation des objets discrets sous forme d'union de boules discrètes (l'axe médian). Cette représentation étant parfois redondante, nous avons tout d'abord montré que la notion d'axe médian minimum entraînait dans une classe de problèmes algorithmiques difficiles (NP-difficiles). Par la suite, plusieurs heuristiques ont été proposées pour réduire cet ensemble sans perte d'information. Enfin, nous nous sommes intéressés à la notion de filtrage (donc avec perte) de l'axe médian. Ces différents outils ont été utilisés dans une problématique de transmission progressive d'objets discrets.

D'un point de vue contractuel, les développements liés à l'arithmétique d'intervalles et aux grilles isothétiques irrégulières se sont déroulés dans le cadre de l'ACI Jeunes Chercheurs GeomDiGIT (voir section 1.6). Concernant l'analyse d'objets fondamentaux, nous rejoignons ici le projet ANR Blanc GeoDib sur lequel nous reviendrons dans les perspectives.

En termes de collaborations, ces éléments ont été très souvent le fruit de collaborations ou de discussions (voir sections 1.4.2 et 1.4.3) et ont donné lieu, en plus de publications, à des développements logiciels diffusés (voir annexe A).

8.8 Perspectives et projets

Si des perspectives ont déjà été données dans les conclusions des différents chapitres, nous revenons plus précisément sur quatre d'entre elles mettant en œuvre plus précisément des collaborations et projets.

DÉFINITION ET ALGORITHMIQUE ASSOCIÉES AUX OBJETS FONDAMENTAUX POUR L'INTÉGRATION D'INCERTITUDE OU DE BRUIT

Une des forces de la géométrie discrète, c'est le fait que l'on puisse faire des calculs exacts et que l'arithmétique sous-jacente nous permet de proposer des algorithmes très efficaces. Une des critiques parfois formulée porte sur le fait qu'elle relègue la gestion de l'incertitude ou du bruit soit en pré, soit en post-traitement. L'un des objectifs du projet ANR Blanc Geodib est de repousser ces limites de la géométrie discrète pour formaliser, arithmétiser et proposer des algorithmes pour intégrer ces incertitudes. Pour cela, il faut revoir les définitions mêmes des objets fondamentaux. Dans ce contexte, nous nous intéresserons plus particulièrement à des définitions arithmétiques intégrant des notions d'*anti-aliasing* ou permettant d'être moins sensibles aux *outliers* (points discrets non représentatifs d'une droite, ces points sont catastrophiques dans la reconnaissance classique).

Ce projet, démarré en septembre 2006, prendra fin en 2010.

RECONSTRUCTION PARALLÈLE DE CONTOURS DISCRETS À BASE DE CARTES COMBINATOIRES

Ce projet correspond à une récente collaboration avec GUILLAUME DAMIAND (SIC, Poitiers) dont les activités principalement portent sur l'utilisation des cartes combinatoires pour la modélisation et la définition d'algorithmes de traitement et d'analyse d'images. L'objectif de ce projet est de guider une simplification de la carte combinatoire associée à une image en intégrant des opérateurs géométriques fondés sur les droites et plans discrets. La structure de la carte offre de plus l'avantage de permettre des traitements parallèles ou hiérarchiques.

Notre participation à ce projet porte sur la définition d'opérateurs rapides d'union de segments de droites discrètes. L'application de tels opérateurs en parallèle permet d'espérer une reconstruction plus globale et moins sujette à un choix de parcours ou un séquencement des opérations.

En dimension 3, les problèmes théoriques sont importants mais les résultats espérés en sont d'autant plus intéressants.

COUVERTURE PAR ARCS DE CERCLE POUR LA DÉFINITION D'ESTIMATEURS GÉOMÉTRIQUES CONVERGENTS

Cette perspective rejoint les premiers développements que nous avons fait au cours de la thèse portant sur la définition d'estimateurs géométriques (longueur, tangente, courbure,...) sur des données discrètes. Nos contributions, basées sur une approche géométrique, voire analytique, ont très vite montré leur limite car certains comportements échappaient à notre formalisation. Par la suite, il s'est avéré qu'une approche arithmétique était beaucoup plus

pertinente [dV07].

Géométriquement, ces approches utilisent la couverture tangentielle d'une courbe discrète (couverture engendrée par les segments discrets maximaux calculés en tous points) et des propriétés de longueur ou de distribution des segments maximaux permettent de prouver la convergence de certains estimateurs. Une couverture intéressante, à la fois d'un point de vue théorique et d'un point de vue pratique, serait celle engendrée par des arcs de cercles maximaux. Beaucoup de questions portent sur la distribution des longueurs des arcs par exemple. Pour avoir des premiers résultats expérimentaux, des efforts restent à faire concernant l'algorithme associé à la reconnaissance de cercles discrets.

AXE MÉDIAN HIÉRARCHIQUE : THE DIGITAL SPHERE TREE

En modélisation ou synthèse d'images, la structure par arbre de sphères connaît une certaine popularité dans la communauté de modélisation pour son usage dans de nombreux domaines : détection de collisions, rendu avec des techniques *point based rendering*, ...

En géométrie discrète, quelques travaux ont été menés pour représenter des objets avec une structure multirésolution (basée sur un amincissement homotope contraint) [PLB01]. Cette structure hiérarchique ne remet pas en cause l'axe médian construit (et donc les rayons des boules) sur le niveau initial, elle ne fait juste que le simplifier.

Ce que nous souhaitons analyser, c'est une structure hiérarchique en intégrant, comme dans le cas continu, une flexibilité quant à la réversibilité de la construction. L'objectif est dans mettre en place des processus de fusion de boules par exemple. Pour arriver à cela, nous devrons sûrement exploiter le diagramme de puissance discret permettant de mieux interpréter les interactions entre boules.

Conclusions et perspectives

Annexe A

Réalisations logicielles et diffusion

Disponibles sur le site du Technical Committee 18 de l'IAPR <http://www.cb.uu.se/~tc18/>, rubrique Code, ou sur <http://liris.cnrs.fr/david.coeurjolly>.

- **libvol**, **liblongvol**, **voltools** : Bibliothèque et outils de manipulation d'objets discrets.
- **volgen** : outils de création d'objet discrets simples (cube, sphère, caténoid,...). Cet outils fournit aussi une signature de l'objet euclidien sous-jacent afin d'évaluer des estimateurs géométrique d'aire, de normale et de courbure.
- **sedt** : implémentation de l'algorithme linéaire en temps de transformation en distance euclidienne [Hir96] (utilisant **libvol** et **liblongvol**).
- **DistanceTransform** : projet contenant les différents codes pour le calcul de la transformée en distance en temps linéaire, la transformée en distance inverse et l'extraction de l'axe médian aussi en temps linéaires [CM07]
- **reconstruction** : (coll. avec LOUTFI ZERRARGA (M2R)) reconstruction réversible dans des 4-courbes simples [CZ06].
- **Digital plane preimage analysis** : outils permettant de construire et de visualiser la préimage d'un plan discret
- Challenge *Comparative Evaluation of Digital Plane Segment Recognition Algorithms* (coordinateur) : infrastructure logicielle et implémentations d'algorithmes de reconnaissance de plans discrets. Ce challenge du TC18-IAPR a pour objectif de mettre en commun les différents développements de plusieurs laboratoires (actuellement LAIC Clermont-Ferrand et LIRIS Lyon).

Bibliographie

- [AAS97] E. Andres, R. Acharya, and C. Sibata. Discrete analytical hyperplanes. *Graphical Models and Image Processing*, 59(5) :302–309, September 1997. [50](#)
- [ACK01] N. Amenta, S. Choi, and R. K. Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry : Theory and Applications*, 19(2-3) :127–153, 2001. [100](#)
- [ACKO88] L. J. Aupperle, H. E. Conn, J. M. Keil, and J. O'Rourke. Covering orthogonal polygons with squares. In *Proceedings of the 26th Annual Allerton Conf. Comm. Control Comput.*, pages 97–106, 1988. [108](#)
- [AE07] D. Attali and H. Edelsbrunner. Inclusion-exclusion formulas from independent complexes. *Discrete & Computational Geometry*, 37(1) :59–77, 2007. [100](#), [115](#), [120](#)
- [AJ97] E. Andrès and M.-A. Jacob. The discrete analytical hyperspheres :. *IEEE Transactions on Visualization and Computer Graphics*, 3(1) :75–86, January/March 1997. [51](#)
- [AM97] D. Attali and A. Montanvert. Computing and simplifying 2d and 3d continuous skeletons. *Computer Vision and Image Understanding*, 67(3) :161–273, 1997. [115](#)
- [And00] E. Andrès. Modélisation analytique discrète d’objets géométriques, 2000. Laboratoire IRCOM-SIC, université de Poitiers, habilitation à diriger des recherches. [34](#), [36](#), [38](#), [118](#)
- [And02] E. Andrès. Defining discrete objects for polygonalization : the standard model. In *10th International Conference on Discrete Geometry for Computer Imagery*, number 2301 in LNCS, pages 313–325. Springer, 2002. [36](#)
- [Att95] D. Attali. *Squelettes et graphes de Voronoi 2D et 3D*. PhD thesis, Université Josphe Fourier - Grenoble 1, 1995. [115](#)
- [Aur87] F. Aurenhammer. Power diagrams : Properties, algorithms, and applications. *SIAM Journal on Computing*, 16(1) :78–96, February 1987. :1987:PDP
- [AŽ95] D. M. Acketa and J. D. Žunić. On the maximal number of edges of convex digital polygons included into a $m \times m$ -grid. *Journal of Combinatorial Theory, Serie A*(69) :358–368, 1995. [53](#)
- [Bár02] I. Bárány. Random points, convex bodies, lattices. In *ICM : Proceedings of the International Congress of Mathematicians*, 2002. [52](#)
- [BB91] A. Balog and I. Bárány. On the convex hull of the integer points in a disc. In Acm, editor, *Proceedings of the 7th Annual Symposium on Computational Geometry (SCG '91)*, pages 162–165, North Conway, NH, USA, June 1991. ACM Press. [53](#)
- [BBN00] R. P. Barneva, V. E. Brimkov, and P. Nehlig. Thin discrete triangular meshes. *Theoretical Computer Science*, 246(1-2) :73–105, 2000. [118](#)
- [BCK07] V. Brimkov, D. Coeurjolly, and R. Klette. Digital planarity - a review. *Discrete Applied Mathematics*, 155(4) :468–495, feb 2007. [25](#), [26](#), [48](#)

BIBLIOGRAPHIE

- [BCO⁺96] J. D. Boissonnat, A. Cerezo, O. Devillers, J. Duquesne, and M. Yvinec. An algorithm for constructing the convex hull of a set of spheres in dimension d. *Computational Geometry : Theory and Applications*, 6(2) :123–130, 1996. [100](#)
- [BD05] V. E. Brimkov and S. S. Dantchev. Complexity analysis for digital hyperplane recognition in arbitrary fixed dimension. In *International Conference on Discrete Geometry for Computer Imagery*, pages 287–298, 2005. [58](#)
- [Ber71] J. Bernoulli. *Recueil pour les astronomes*. 1771. [13](#)
- [BFJP07] V. Berthé, C. Fiorio, D. Jamet, and F. Philippe. On some applications of generalized functionality for arithmetic discrete planes. *Image and Vision Computing*, 25(10) :1671–1684, October 2007. [50](#)
- [BGKW95] H. Breu, J. Gil, D. Kirkpatrick, and M. Werman. Linear time euclidean distance transform algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5) :529–533, 1995. [84](#)
- [BHL92] I. Bárány, R. Howe, and L. Lovász. On integer points in polyhedra : A lower bound. *Combinatorica*, 12, 1992. [52](#)
- [BL98] I. Bárány and D. G. Larman. The convex hull of the integer points in a large ball. *Math. Annalen*, 312 :167–181, 1998. [53](#)
- [Blu67] H. Blum. A transformation for extracting descriptors of shape. In *Models for the Perception of Speech and Visual Forms*, pages 362–380. MIT Press, 1967. :1964
- [BMAVH94] F. Benhamou, D. Mc Allester, and P. Van Hentenryck. CLP(intervals) Revisited. In *Proceedings of the 1994 International Symposium on Logic Programming*, pages 124–138. MIT Press, 1994. [41](#)
- [BN95] G. Borgefors and I. Nyström. Quantitative shape analysis of volume images – reducing the set of centres of maximal spheres. In *Proc. SSAB Symposium on Image Analysis*, pages 5–8, Linköping, Sweden, 1995. [108](#)
- [BN97] G. Borgefors and I. Nyström. Efficient shape representation by minimizing the set of centers of maximal discs/spheres. *Pattern Recognition Letters*, 18 :465–472, 1997. [108](#)
- [BO02] G. Bradshaw and C. O’Sullivan. Sphere-tree construction using dynamic medial axis approximation. In *SCA ’02 : Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 33–40, New York, NY, USA, 2002. ACM Press. [120](#)
- [Bor86] G. Borgefors. Distance transformations in digital images. *Computer Vision, Graphics, and Image Processing*, 34(3) :344–371, June 1986. [28](#), [84](#), [85](#), [86](#)
- [Bre77] J. E. Bresenham. A linear algorithm for incremental digital display of circular arcs. *Communications of the ACM*, 20(2) :100–106, February 1977. [51](#)
- [Buz03] L. Buzer. A linear incremental algorithm for naive and standard digital lines and planes recognition. *Graphical Models*, 65(1-3) :61–76, May 2003. [55](#)
- [Buz07] L. Buzer. Computing multiple convex hulls of a simple polygonal chain in linear time. In *23rd European Workshop on Computational Geometry*, 2007. [54](#), [63](#)
- [BY95] J.-D. Boissonnat and M. Yvinec. *Géométrie algorithmique*. Ediscience International, 1995. [54](#), [57](#)
- [BYBH96] R. Bar-Yehuda and E. Ben-Hanoch. A linear time algorithm for covering simple polygons with similar rectangles. *Intern. J. of Computational Geometry and Applications*, 6(1) :79–102, 1996. [108](#)
- [CB06] D. Coeurjolly and V. Brimkov. Computational aspects of digital plane and hyperplane recognition. In *11th International Workshop on Combinatorial Image Analysis*, volume 4040 of *Lecture Notes in Computer Science*, pages 291–306. Springer-Verlag, 2006. invited talk. [57](#), [59](#)

- [CCZ07] M. Couplie, D. Coeurjolly, and R. Zrour. Discrete bisector function and euclidean skeleton in 2d and 3d. *Image and Vision Computing*, 25(10) :1519–1698, oct 2007. [x](#), [86](#), [89](#), [90](#), [93](#), [94](#), [115](#)
- [CDJS06] D. Coeurjolly, F. Dupont, L. Jospin, and I. Sivignon. Optimization schemes for the reversible discrete volume polyhedrization using marching cubes simplification. In *13th International Conference on Discrete Geometry for Computer Imagery*, volume 4245 of *LNCS*, pages 413–424, Szeged, Hungary, 2006. Springer-Verlag. [27](#), [76](#), [77](#)
- [CGRT04] D. Coeurjolly, Y. Gerard, J. P. Reveillès, and L. Tougne. An elementary algorithm for digital arc segmentation. *Discrete Applied Mathematics*, 139(1-3) :31–50, 2004. [54](#), [55](#), [61](#), [62](#), [63](#), [64](#)
- [CGS04] D. Coeurjolly, A. Guillaume, and I. Sivignon. Reversible discrete volume polyhedrization using marching cubes simplification. In *SPIE Vision Geometry XII*, volume 5300, pages 1–11, San Jose, USA, 2004. [73](#), [74](#), [75](#)
- [Cha93] B. Chazelle. An optimal convex hull algorithm in any fixed dimension. *Discrete & Computational Geometry*, 10 :377–409, 1993. [54](#)
- [Cha96] T.M. Chan. Optimal output-sensitive convex hull algorithms in two and three dimensions. *GEOOMETRY : Discrete & Computational Geometry*, 16, 1996. [54](#)
- [Chr75] E. B. Christoffel. Observatio arithmetica. *Annali di Mathematica*, 6 :148–152, 1875. 2nd series. [13](#)
- [CHS07] D. Coeurjolly, J. Hulin, and I. Sivignon. Finding a minimum medial axis of a discrete shape is np-hard. Technical Report RR-2007-026, Laboratoire LIRIS, UMR-CNRS 5205, Université Claude Bernard Lyon 1, 2007. [109](#), [111](#), [112](#), [121](#)
- [CLR90] T. Cormen, C. Leiserson, and R. Rivest. *Introduction à l’algorithmique*. Dunod, 1990. [28](#), [29](#), [113](#), [114](#)
- [CM91] J. M. Chassery and A. Montanvert. *Géométrie discrète en analyse d’images*. Hermès Paris, 1991. [36](#)
- [CM99] O. Cuisenaire and B. Macq. Fast euclidean distance transformations by propagation using multiple neighbourhoods. *Computer Vision and Image Understanding*, 76(2) :163–172, November 1999. [84](#)
- [CM07] D. Coeurjolly and A. Montanvert. Optimal separable algorithms to compute the reverse euclidean distance transformation and discrete medial axis in arbitrary dimension. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3) :437–448, mar 2007. [85](#), [95](#), [97](#), [98](#), [99](#), [100](#), [115](#)
- [CMC07] D. Coeurjolly, A. Montanvert, and J. M. Chassery, editors. *Géométrie discrète et images numériques*. Traité IC2, Signal et Image. Hermès Paris, France, sep 2007. [13](#), [23](#), [73](#), [84](#), [85](#), [95](#), [96](#), [98](#)
- [Coe02] D. Coeurjolly. *Algorithmique et géométrie discrète pour la caractérisation des courbes et des surfaces*. PhD thesis, Université Lumière Lyon 2, Bron, Laboratoire ERIC, dec 2002. [49](#), [57](#), [58](#), [89](#)
- [Coe05] D. Coeurjolly. Supercover model and digital straight line recognition on irregular isothetic grids. In E. Andres, G. Damiand, and P. Lienhardt, editors, *12th International Conference on Discrete Geometry for Computer Imagery*, volume 3429 of *Lecture Notes in Computer Science*, pages 311–322. Springer-Verlag, 2005. [66](#)
- [Coo71] S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, pages 151–158, New York, 1971. [:1971](#)
- [CPW93] C. H. Chen, L. F. Pau, and P. S. Wang, editors. *Segmentation tools in Mathematical Morphology*. World Scientific, 1993. [84](#)

BIBLIOGRAPHIE

- [CSD⁺05] D. Coeurjolly, I. Sivignon, F. Dupont, F. Feschet, and J. M. Chassery. On digital plane preimage structure. *Discrete Applied Mathematics*, 151(1–3) :78–92, 2005. [57](#)
- [CTK00] J. H. Chuang, C. H. Tsai, and M. C. Ko. Skeletonization of three-dimensional object using generalized potential field. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(11) :1241–1251, November 2000. [85](#)
- [CZ06] D. Coeurjolly and L. Zerarga. Supercover model, digital straight line recognition and curve reconstruction on the irregular isothetic grids. *Computer and Graphics*, 30(1) :46–53, 2006. [66](#), [67](#)
- [Dan80] P. E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14(3) :227–248, November 1980. [84](#)
- [dB94] G. Sammiti di Baja. Well-shaped, stable, and reversible skeletons from the (3,4)-distance transform. *J. Visual Communication and Image Representation*, 5 :107–115, 1994. [85](#)
- [dBvKOS00] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry*. Springer-Verlag, 2000. [86](#), [89](#)
- [DCA06] M. Dexet, D. Coeurjolly, and E. Andres. Invertible polygonalization of 3d planar digital curves and application to volume data reconstruction. In *International Symposium on Visual Computing (ISVC)*, volume 4292 of *LNCS*, Lake Tahoe, Nevada, USA, 2006. [27](#)
- [Deb95] I. Debled-Rennesson. *Etude et reconnaissance des droites et plans discrets*. PhD thesis, Université Louis Pasteur, 1995. [48](#), [55](#), [56](#), [118](#)
- [Dex06] M. Dexet. *Architecture d'un modeleur géométrique à base topologique d'objets discrets et méthodes de reconstruction en dimensions 2 et 3*. Thèse de doctorat, Université de Poitiers, 2006. [27](#), [57](#), [66](#), [70](#), [77](#)
- [DGB03] F. Dupont, B. Gilles, and A. Baskurt. Lossless and scalable 3d obbject coding method based on medial axis transformation. In *PCS'03 (Picture Coding Symposium)*, 2003. [117](#)
- [DGG03] T. Dey, J. Giesen, and S. Goswami. Shape segmentation and matching with flow discretization. pages 25–36. 2003. [84](#), [89](#)
- [DK83] D. P. Dobkin and D. G. Kirkpatrick. Fast detection of polyhedral intersection. *Theoretical Computer Science*, 27(3) :241–253, December 1983. : DobkinK1983
- [dPP90] H. de Fraysseix, J. Pach, and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10 :41–51, 1990. [30](#)
- [DR95] I. Debled Rennesson and J.-P. Reveillès. A linear algorithm for segmentation of digital curves. In *International Journal on Pattern Recognition and Artificial Intelligence*, volume 9, pages 635–662, 1995. [55](#)
- [DS84] L. Dorst and A. N. M. Smeulders. Discrete representation of straight lines. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6 :450–463, 1984. [49](#)
- [DS91] L. Dorst and A. W. M. Smeulders. Decomposition of discrete curves into piecewise straight segments in linear time. In *Contemporary Mathematics*, volume 119, 1991. [55](#)
- [dV07] F. de Vieilleville. *Analyse des parties linéaires des objets discrets pour l'extraction de caractéristiques géométriques*. Thèse de doctorat, Université de Bordeaux 1, 2007. [53](#), [55](#), [63](#), [78](#), [127](#)
- [Fei98] U. Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4) :634–652, 1998. [113](#)

- [FM05] C. Fouard and G. Malandain. 3-d chamfer distances and norms in anisotropic grids. *Image and Vision Computing*, 23(2) :143–158, February 2005. :ivc :2005
- [FST96] J. Françon, J. M. Schramm, and M. Tajine. Recognizing arithmetic straight lines and planes. In *6th Discrete Geometry for Computer Imagery*, volume 1176 of *LNCS*, pages 141–150. Springer-Verlag, 1996. [58](#)
- [FT99] F. Feschet and L. Tougne. Optimal time computation of the tangent of a discrete curve : Application to the curvature. In *8th International Workshop in Discrete Geometry for Computer Imagery*, pages 31–40. Springer-Verlag, LNCS, 1568, 1999. [63](#), [78](#)
- [FT06] C. Fiorio and J.-L. Toutant. Arithmetic discrete hyperspheres and separatin-gness. In Attila Kuba, László G. Nyúl, and Kálmán Palágyi, editors, *DGCI*, volume 4245 of *Lecture Notes in Computer Science*, pages 425–436. Springer, 2006. [51](#)
- [FTJ06] C. Fiorio, J.-L. Toutant, and D. Jamet. Discrete Circles : an arithmetical approach with non-constant thickness. In *Vision Geometry XIV, IS&T/SPIE 18th Annual Symposium Electronic Imaging*, pages 60660C01–60660C12, 01 2006. :2006 :LIRMM-00128281 :1
- [GDRZ05] Y. Gerard, I. Debled-Rennesson, and P. Zimmermann. An elementary digital plane recognition algorithm. *DAMATH : Discrete Applied Mathematics and Combinatorial Operations Research and Computer Science*, 151, 2005. [56](#), [57](#), [58](#)
- [GF96] Y. Ge and J. M. Fitzpatrick. On the generation of skeletons from discrete euclidean distance maps. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 18(11) :1055–1066, 1996. [85](#)
- [GJ79] M. R. Garey and D. S. Johnson. *Computers and intractability ; a guide to the theory of NP-completeness*. W.H. Freeman, 1979. [52](#)
- [GK03] Peter Giblin and Benjamin B. Kimia. On the local form and transitions of symmetry sets, medial axes, and shocks. *International Journal of Computer Vision*, 54(1-3) :143–157, 2003. [85](#)
- [GK04] P. Giblin and B. B. Kimia. A formal classification of 3d medial axis points and their local geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2) :238–251, 2004. :677854
- [GL95] C. Gotsman and M. Lindenbaum. Euclidean voronoi labelling on the multidimensional grid. *Pattern Recognition Letters*, 16 :409–415, 1995. [84](#)
- [GM98] W. Guan and S. Ma. A list-processing approach to compute voronoi diagrams and the euclidean distance transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(7) :757–761, 1998. [84](#)
- [GO97] J. E. Goodman and J. O'Rourke, editors. *Handbook of Discrete and Computational Geometry*. CRC Press, 1997. [54](#), [60](#), [108](#)
- [Gol95] M. Goldwasser. A survey of linear programming in randomized subexponential time. *SIGACTN : SIGACT News (ACM Special Interest Group on Automata and Computability Theory)*, 26, 1995. [52](#)
- [Gra72] R. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, 1 :132–133, 1972. :72
- [GS87] B. Grunbaum and G. C. Shephard. *Tilings and Patterns*. Freeman, San Francisco, CA, 1987. [19](#)
- [Har89] J. Hartong. Une théorie du continu. In *La mathématique non standard*, Fondements des Sciences. Editions du CNRS, Paris, 1989. [14](#)
- [Har98] S. Har-Peled. An output sensitive algorithm for discrete convex hulls. *Computational Geometry. Theory and Applications*, 10(2) :125–138, May 1998. [54](#)

- [Hav01] V. Havran. *Heuristic Ray Shooting Algorithms*. PhD thesis, Czech Technical University, Faculty of Electrical Engineering, Department of Computer Science and Engineering, 2001. [20](#)
- [Her98] G. T. Herman. *Geometry of digital spaces*. Birkhäuser, Boston, 1998. [24](#)
- [Hir96] T. Hirata. A unified linear-time algorithm for computing distance maps. *Information Processing Letters*, 58(3) :129–133, May 1996. [84](#), [86](#), [129](#)
- [HT06] X. Hilaire and K. Tombre. Robust and accurate vectorization of line drawings. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(6) :890–904, 2006. [27](#)
- [HVR05] W. H. Hesselink, M. Visser, and J. B. T. M. Roerdink. Euclidean skeletons of 3d data sets in linear time by the integer medial axis transform. In C. Ronse, L. Najman, and E. Decencière, editors, *Mathematical Morphology : 40 Years On (Proc. 7th Intern. Symp. on Mathematical Morphology)*, pages 259–268. Springer, April 2005. [89](#)
- [HW75] G. H. Hardy and E. M. Wright. *Introduction to the theory of numbers*. Oxford press, 1975. [61](#), [62](#), [90](#)
- [Jar73] R. Jarvis. On the identification of the convex hull of a finite set of points in the plane. *Information Processing Letters*, 2 :18–21, 1973. [54](#)
- [JK97] A. Jonas and N. Kiryati. Digital representation schemes for 3D curves. *Pattern Recognition*, 30(11) :1803–1816, 1997. [34](#)
- [JM95] K. Jansen and H. Müller. The minimum broadcast time problem for several processor networks. *Theoretical Computer Science*, 147(1–2) :69–85, 7 August 1995. [:JansenM1995](#) :[69](#)
- [Jos06] L. Jospin. Transmission progressive d’images 3d à partir du calcul de l’axe médian. Rapport master recherche, Ecole Normale Supérieure de Lyon, Laboratoire LIRIS, 2006. <http://liris.cnrs.fr/publis?id=2983>. [119](#)
- [Kim84] C. E. Kim. Three-dimensional digital planes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6 :639–645, 1984. [56](#), [58](#)
- [Kov90] V. A. Kovalevsky. New definition and fast recognition of digital straight segments and arcs. In *Proc. 10th Intl. Conf. on Pattern Recognition*, pages 31–34, 1990. [55](#)
- [KR04a] R. Klette and A. Rosenfeld. *Digital Geometry : Geometric Methods for Digital Picture Analysis*. Series in Computer Graphics and Geometric Modelin. Morgan Kaufmann, 2004. [34](#), [48](#)
- [KR04b] R. Klette and A. Rosenfeld. Digital straightness—a review. *Discrete Applied Mathematics*, 139(1-3) :197–230, April 2004. [25](#), [26](#), [48](#)
- [KS91] C. E. Kim and I. Stojmenovic. On the recognition of digital planes in three-dimensional space. *Pattern Recognition Letters*, 12(11) :665–669, 1991. [56](#), [58](#)
- [KSKB95] R. Kimmel, D. Shaked, N. Kiryati, and A. M. Bruckstein. Skeletonization via distance maps and level sets. *Computer Vision and Image Understanding*, 62(3) :381–391, 1995. [85](#)
- [Lac96] J.-O. Lachaud. Topologically defined isosurfaces. In *6th Discrete Geometry for Computer Imagery*, pages 245–256. LNCS 1176, Springer-Verlag, 1996. [73](#)
- [Lat91] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, 1991. [84](#)
- [LB93] M. Lindenbaum and A. Bruckstein. On recursive, $O(n)$ partitioning of a digitized curve into digital straigth segments. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(9) :949–953, September 1993. [49](#), [55](#)

- [LC87] W. E. Lorensen and H. E. Cline. Marching cubes : a high resolution 3D surface construction algorithm. In M. C. Stone, editor, *SIGGRAPH '87 Conference Proceedings (Anaheim, CA, July 27–31, 1987)*, pages 163–170. Computer Graphics, Volume 21, Number 4, July 1987. [27](#), [73](#)
- [LCG98] L. J. Latecki, C. Conrad, and A. Gross. Preserving topology by a digitization process. *Journal of Mathematical Imaging and Vision*, 8(2) :131–159, March 1998. [66](#)
- [Lie04] A. Lieutier. Any open bounded subset has the same homotopy type as its medial axis. *Computer-Aided Design*, 36(11) :1029–1046, September 2004. [96](#)
- [LL92] F. F. Leymarie and M. D. Levine. Simulating the grassfire transform using an active contour model. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 14(1) :56–75, 1992. [85](#)
- [LM00] J.-O. Lachaud and A. Montanvert. Continuous analogs of digital boundaries : A topological approach to iso-surfaces. *Graphical Models and Image Processing*, 62 :129–164, 2000. [27](#), [73](#)
- [LW03] C. Lincke and C. A. Wüthrich. Surface digitizations by dilations which are tunnel-free. *Discrete Applied Mathematics*, 125(1) :81–91, 2003. [38](#)
- [Mat88] G. Matheron. Examples of topological properties of skeletons. In J. Serra, editor, *Image Analysis and Mathematical Morphology, Volume 2 : Theoretical Advances*, pages 217–238. Academic Press, Londres, 1988. [96](#)
- [Mci85] M. D. Mcilroy. A note on discrete representation of lines. *AT&T Technical Journal*, 64(2) :481–490, February 1985. [49](#)
- [MDKC00] J. Mukherjee, P. P. Das, M. Aswatha Kumarb, and B. N. Chatterjib. On approximating euclidean metrics by digital distances in 2d and 3d. *Pattern Recognition Letters*, 21(6–7) :573–582, 2000. [84](#)
- [Meg84a] N. Megiddo. Linear programming in linear time when the dimension is fixed. *JACM : Journal of the ACM*, 31, 1984. [52](#), [58](#)
- [Meg84b] N. Megiddo. Linear programming in linear time when the dimension is fixed. *Journal of the ACM*, 31(1) :114–127, January 1984. [61](#)
- [Meg84c] Nimrod Megiddo. Linear programming in linear time when the dimension is fixed. *J. ACM*, 31(1) :114–127, January 1984. [55](#)
- [Mel87] A. A. Melkman. On-line construction of the convex hull of a simple polyline. *Information Processing Letters*, 25(1) :11–12, April 1987. [:Melkman1987](#)
- [Mes02] M. M. Mesmoudi. A simplified recognition algorithm of digital planes pieces. In *10th International Conference, Discrete Geometry for Computer Imagery*, volume 2301 of *LNCS*, pages 404–416, Bordeaux, France, April 2002. Springer Verlag. [58](#)
- [Mon69] U. Montanari. Continuous skeletons from digitized images. *Journal of the Association for Computing Machinery*, 16(4) :534–549, oct 1969. [84](#)
- [Moo66] R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, N.J., 1966. [39](#), [40](#)
- [Moo79] R. E. Moore. *Methods and Applications of Interval Arithmetic*. Studies in Applied Mathematics. SIAM, Philadelphia, 1979. [39](#), [40](#)
- [MQR03] C. R. Maurer, R. Qi, and V. Raghavan. A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2) :265–270, February 2003. [84](#), [89](#)
- [MRH00] A. Meijster, J. B. T. M. Roerdink, and W. H. Hesselink. A general algorithm for computing distance transforms in linear time. In L. Vincent J. Goutsias and D. S. Bloomberg, editors, *Mathematical Morphology and its Applications to Image and Signal Processing*, pages 331–340. Kluwer, 2000. [84](#), [86](#)

BIBLIOGRAPHIE

- [Mul92] J. C. Mullikin. The vector distance transform in two and three dimensions. *Computer Vision, Graphics, and Image Processing. Graphical Models and Image Processing*, 54(6) :526–535, November 1992. [84](#)
- [Nag05] B. Nagy. A comparison among distances based on neighborhood sequences in regular grids. In *Image Analysis, 14th Scandinavian Conference*, number 3540 in LNCS, pages 1027–1036. Springer, 2005. [84](#)
- [ND97] F. Nilsson and P.-E. Danielsson. Finding the minimal set of maximum disks for binary objects. *Graphical Models and Image Processing*, 59(1) :55–60, January 1997. [:1997 :FMS](#)
- [NY06] T. S. Newman and H. Yi. A survey of the marching cubes algorithm. *Computers & Graphics*, 30(5) :854–879, October 2006. [27](#), [78](#)
- [Ogn94] R. Ogniewicz. A multiscale MAT from Voronoi diagrams : the skeleton-space and its application to shape description and decomposition. In C. Arcelli et al., editors, *Aspects of Visual Form Processing*, pages 430–439. World Scientific, Singapore, 1994. [84](#)
- [OK95] D. Cohen Or and A. Kaufman. Fundamentals of surface voxelization. *Graphical Models and Image Processing*, 57(6) :453–461, November 1995. [34](#), [36](#)
- [OKM86] J. O'Rourke, S. R. Kosaraju, and N. Megiddo. Computing circular separability. *Discrete and Computational Geometry*, 1(1) :105–113, 1986. [:1986a](#)
- [O'R81] J. O'Rourke. An on-line algorithm for straight lines between data ranges. *Communication of the ACM : Programming Techniques and Data Structures*, 24(9) :574–578, 1981. [55](#)
- [Pio99] S. Pion. *De la géométrie algorithmique au calcul géométrique*. Thèse de doctorat, Université Nice Sophia-Antipolis, 1999. [38](#)
- [PLB01] S. Prevost, L. Lucas, and E. Bittar. Multiresolution and shape optimization of implicit skeletal model. In V. Skala, editor, *WSCG 2001 Conference Proceedings*, 2001. [127](#)
- [PS85] F. P. Preparata and M. I. Shamos. *Computational Geometry : An Introduction*. Springer-Verlag, 1985. [54](#), [55](#), [86](#), [89](#)
- [Rag90] I. Ragnemalm. *Contour processing distance transforms*, pages 204–211. World Scientific, 1990. [84](#)
- [Rag93] I. Ragnemalm. *The Euclidean Distance Transform*. PhD thesis, Linköping University, Linköping, Sweden, 1993. [85](#)
- [RB93] I. Ragnemalm and G. Borgefors. *The Euclidean Distance Transform*, chapter Towards a minimal shape representation using maximal discs, pages 245–260. Linköping Studies in Science and Technology. Dissertations No. 304., Linköping University, April 1993. [xi](#), [108](#), [113](#), [114](#)
- [Rev91] J. P. Reveillès. *Géométrie discrète, calcul en nombres entiers et algorithmique*. PhD thesis, Université Louis Pasteur - Strasbourg, 1991. [14](#), [48](#)
- [RP66] A. Rosenfeld and J. L. Pfaltz. Sequential operations in digital picture processing. *Journal of the ACM*, 13(4) :471–494, October 1966. [84](#), [85](#)
- [RP68] A. Rosenfeld and J. L. Pfaltz. Distance functions on digital pictures. *Pattern Recognition*, 1 :33–61, 1968. [84](#)
- [RR84] H. Ratschek and J. Rokne. *Computer Methods for the Range of Functions*. Ellis Horwood Ltd., Chichester, 1984. [:RatschekRokne](#) :84
- [RS97] M. Rueher and C. Solnon. Concurrent cooperating solvers over reals. *Reliable Computing*, 3(3) :325–333, 1997. [41](#)
- [RT00a] E. Remy and E. Thiel. Optimizing 3D chamfer masks with norm constraints. In *Int. Workshop on Combinatorial Image Analysis*, pages 39–56, Caen, July 2000. [84](#)

- [RT00b] C. Ronse and M. Tajine. Discretization in hausdorff space. *Journal of Mathematical Imaging & Vision*, 12(3) :219–242, June 2000. [34](#)
- [RT02] E. Remy and E. Thiel. Medial axis for chamfer distances : computing look-up tables and neighbourhoods in 2D or 3D. *Pattern Recognition Letters*, 23(6) :649–661, 2002. [85](#)
- [RT05] E. Remy and E. Thiel. Exact medial axis with euclidean distance. *Image and Vision Computing*, 23(2) :167–175, 2005. [85](#)
- [Sam90] H. Samet. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley Reading MA, 1990. [20](#), [66](#)
- [SBTZ99] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. Zucker. The hamilton-jacobi skeleton. In *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV-99)*, volume II, pages 828–834, Los Alamitos, CA, September 20–27 1999. IEEE. [85](#)
- [SC03] I. Sivignon and D. Coeurjolly. From digital plane segmentation to polyhedral representation. In *Theoretical Foundations of Computer Vision "Geometry, Morphology, and Computational Imaging"*, number 2616 in LNCS, Springer-Verlag, pages 356–367, 2003. [26](#)
- [SC06] I. Sivignon and D. Coeurjolly. Minimal decomposition of a digital surface into digital plane segments is np-hard. In *13th International Conference on Discrete Geometry for Computer Imagery*, number 4245 in LNCS, pages 674–685, Szeged, Hungary, 2006. Springer-Verlag. [27](#), [30](#), [74](#)
- [Sch86] A. Schrijver. *Theory of Linear and Integer Programming*. Wiley and Sons, 1986. [52](#)
- [SDC03] I. Sivignon, F. Dupont, and J. M. Chassery. Decomposition of a three-dimensional discrete object surface into discrete plane pieces. *Algorithmica*, 38(1) :25–43, 2003. [26](#), [77](#)
- [SDC04] Isabelle Sivignon, Florent Dupont, and Jean-Marc Chassery. Digital intersections : minimal carrier, connectivity, and periodicity properties. *Graphical Models*, 66(4) :226–244, July 2004. [25](#)
- [SDC05] I. Sivignon, F. Dupont, and J. M. Chassery. Reversible polygonalization of a 3D planar discrete curve : Application on discrete surfaces. In *International Conference on Discrete Geometry for Computer Imagery*, pages 347–358, 2005. [26](#)
- [Ser82] J. Serra. *Image Analysis and mathematical Morphology*. Academic Press, London, 1982. [37](#)
- [SHB99] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. PWS Publishing, Pacific Grove, second edition, 1999. [84](#)
- [Siv04] I. Sivignon. *De la caractérisation des primitives ?a reconstruction polyhédrique de surfaces en géométrie discrète*. Thèse de doctorat, Laboratoire des Images et des Signaux, INPG, Grenoble, 2004. [26](#), [57](#), [74](#), [77](#)
- [SK05] P. Stelldinger and U. Köthe. Towards a general sampling theory for shape preservation. *Image and Vision Computing*, 23(2) :237–248, February 2005. [34](#), [66](#)
- [Sny92] J. M. Snyder. Interval analysis for computer graphics. *Computer Graphics*, 26(2), July 1992. Proceedings SIGGRAPH'92. [ix](#), [41](#), [66](#), [69](#)
- [Soi99] P. Soille. *Morphological Image Analysis*. Springer-Verlag, Berlin, Heidelberg, New York, Etats-unis, 1999. [37](#), [104](#)
- [SS02] S. Svensson and G. Sanniti di Baja. Using distance transforms to decompose 3d discrete objects. *Image Vision & Computing*, 20(8) :529–540, June 2002. [84](#)

- [ST91] I. Stojmenović and R. Tosić. Digitization schemes and the recognition of digital straight lines, hyperplanes and flats in arbitrary dimensions. In *Vision Geometry, contemporary Mathematics Series*, volume 119, pages 197–212, American Mathematical Society, Providence, RI, 1991. [50](#), [56](#), [58](#)
- [ST94a] T. Saito and J. I. Toriwaki. New algorithms for Euclidean distance transformations of an n -dimensional digitized picture with applications. *Pattern Recognition*, 27 :1551–1565, 1994. [:94](#)
- [ST94b] T. Saito and J.-I. Toriwaki. Reverse distance transformation and skeletons based upon the euclidean metric for n -dimensionnal digital pictures. *IECE Trans. Inf. & Syst.*, E77-D(9) :1005–1016, September 1994. [85](#), [97](#)
- [Tam87] R. Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM J. Comput.*, 16(3) :421–444, June 1987. [xi](#), [109](#), [111](#)
- [Thi94] E. Thiel. *Les distances de chanfrein en analyse d’images : fondements et applications*. Thèse de doctorat, Université Joseph Fourier, 1994. [117](#)
- [Thi01] E. Thiel. *Géométrie des distances de chanfrein*. Habilitation à Diriger des Recherches, université de la Méditerranée, Aix-Marseille 2, December 2001. <http://www.lif-sud.univ-mrs.fr/~thiel/hdr>. [28](#)
- [TR02] M. Tajine and C. Ronse. Topological properties of hausdorff discretization, and comparison to other discretization schemes. *Theoretical Computer Science*, 283(1) :243–268, June 2002. [34](#)
- [VC00] J. Vittone and J. M. Chassery. Recognition of digital naive planes and polyhedrization. In *9th Discrete Geometry for Computer Imagery*, volume 1953 of *LNCS*, pages 296–307. Springer, 2000. [57](#), [58](#)
- [VCT06a] A. Vacavant, D. Coeurjolly, and L. Tougne. Dynamic reconstruction of complex planar objects on irregular isothetic grids. In *International Symposium on Visual Computing (ISVC)*, number 4292 in *LNCS*, Nevada, USA, 2006. [69](#)
- [VCT06b] A. Vacavant, D. Coeurjolly, and L. Tougne. Topological and geometrical reconstruction of complex objects on irregular isothetic grids. In *13th International Conference on Discrete Geometry for Computer Imagery*, volume 4245 of *LNCS*, Szeged, Hungary, 2006. Springer-Verlag. [68](#), [70](#)
- [VCT07] A. Vacavant, D. Coeurjolly, and L. Tougne. A framework for dynamic implicit curve approximation by an irregular discrete approach. Technical Report RR-LIRIS-2007-020, LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/Ecole Centrale de Lyon, 2007. [70](#)
- [Vee93] P. Veelaert. On the flatness of digital hyperplanes. *Journal of Mathematical Imaging and Vision*, 3 :205–221, 1993. [50](#), [57](#)
- [Vee94] P. Veelaert. Digital planarity of rectangular surface segments. *IEEE Pattern Analysis and Machine Intelligence*, 16(6) :647–652, June 1994. [58](#)
- [Vee99] P. Veelaert. Geometric constructions in the digital plane. *Journal of Mathematical Imaging and Vision*, 11 :99–118, 1999. [25](#)
- [Vee02] P. Veelaert. Concurrency of line segments in uncertain geometry. volume 2301, pages 289–300, Bordeaux, France, 2002. Springer-Verlag. [25](#)
- [Vit99] J. Vittone. *Caractérisation et reconnaissance de droites et de plans en géométrie discrète*. PhD thesis, Université Joseph Fourier-Grenoble 1, décembre 1999. [57](#)
- [Vos93] K. Voss. *Discrete Images, Objects and Functions in \mathbb{Z}^n* . Number 11 in Algorithms and Combinatorics. Springer-Verlag, 1993. [54](#), [63](#)
- [Zol00] N. Y. Zolotykh. On the number of vertices in integer linear programming problems. Technical report, University of Nizhni Novgorod, 2000. [52](#)