



Reconstruction 3D d'un objet compact en tomographie

Charles Soussen

► To cite this version:

Charles Soussen. Reconstruction 3D d'un objet compact en tomographie. Autre. Université Paris Sud - Paris XI, 2000. Français. NNT: . tel-00004126v2

HAL Id: tel-00004126

<https://theses.hal.science/tel-00004126v2>

Submitted on 29 Jan 2004

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

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

ORSAY

n° d'ordre : 6410

UNIVERSITÉ DE PARIS-SUD

CENTRE D'ORSAY

THÈSE

présentée pour obtenir

Le GRADE de DOCTEUR EN SCIENCES

DE L'UNIVERSITÉ PARIS XI ORSAY

par

Charles SOUSSEN

TITRE : RECONSTRUCTION 3D D'UN OBJET COMPACT EN TOMOGRAPHIE

Soutenue le **20 Décembre 2000** devant la commission d'examen

MM.	Bernard CHALMOND	<i>Rapporteur</i>
	Jean-Louis COATRIEUX	<i>Président</i>
	Jean-Marc DINTEN	
MME	Line GARNERO	<i>Rapporteur</i>
MM.	Ali MOHAMMAD-DJAFARI	<i>Directeur de thèse</i>
	Kenneth SAUER	

REMERCIEMENTS

Je remercie tout d'abord Pierre BERTRAND, ancien responsable du DEA ATS, pour m'avoir permis de bénéficier d'une bourse doctorale sans laquelle ma thèse n'aurait pu avoir lieu. Ce travail s'est déroulé au sein du « Groupe Problèmes Inverses » (GPI) du Laboratoire des Signaux et Systèmes (L2S). Je souhaite remercier Guy DEMOMENT, fondateur du GPI et actuel directeur du L2S, pour son accueil dans un environnement aussi favorable.

Ali MOHAMMAD-DJAFARI a dirigé mon travail pendant ces trois années, tout en me laissant une grande liberté. Je tiens à le remercier pour la confiance qu'il a su m'accorder. Durant ma première année de thèse, il m'a donné l'opportunité de le suivre à l'Université de Notre Dame aux États-Unis. Je remercie vivement Ken SAUER, professeur au sein du Département de Génie Électrique de cette université, pour m'avoir accueilli pendant sept mois dans des conditions aussi propices à la recherche. Outre l'intérêt scientifique qu'il a porté à ma thèse, Ken SAUER m'a également fait l'honneur de faire partie du jury. Je le remercie de cette participation d'outre Atlantique!

Bernard CHALMOND et Line GARNERO ont accepté de rapporter sur mon travail. Je leur exprime toute ma gratitude pour le temps qu'ils y ont consacré, et notamment pour l'examen de certaines annexes *compactes* du manuscrit. Leurs commentaires ont permis d'étoffer la bibliographie et d'améliorer largement la lisibilité de certains passages. Je souhaite remercier Jean-Louis COATRIEUX pour l'intérêt qu'il a porté à mon travail depuis le GRETSI'99, et pour avoir accepté de présider le jury. Enfin, j'exprime ma reconnaissance à Jean-Marc DINTEN pour sa participation au jury, sa disponibilité et les perspectives appliquées qu'il a dégagées de mon étude.

La majeure partie de ma thèse s'est déroulée au sein du GPI. J'ai pu bénéficier de nombreux contacts, à la fois scientifiques et humains, avec les membres de cette équipe. Je tiens à remercier particulièrement Jérôme IDIER, source inépuisable de bonnes idées, pour l'intérêt qu'il a porté à ma thèse. Ses conseils avisés ont permis le développement d'un grand nombre de problématiques abordées dans ce manuscrit. Je remercie Jean-François GIOVANNELLI pour les nombreux conseils stratégiques qu'il a pu me donner, en particulier à l'abord de la dernière ligne droite de mon travail. J'en profite pour rendre hommage aux concepts qui lui sont chers : la *carréification*, la thèse sur articles, la programmation sans bugs, *etc.* . .

Je tiens à remercier particulièrement Philippe CIUCIU pour l'ensemble des conseils et des coups de mains qu'il m'a apportés. Philippe et moi avons terminé nos thèses à quelques mois d'intervalle, et par là même, traversé ensemble les perturbations cyclothymiques inhérentes à la fin de thèse. Outre les grands *môments* de nocturnes que nous avons partagés, Philippe n'a pas compté les heures passées à relire de nombreuses parties de ce manuscrit, dont certaines très techniques. À mon sens, ses conseils judicieux ont largement amélioré la qualité du document, à la fois au niveau de son organisation et de sa clarté, et ainsi permis à ma thèse d'*aller plus haut*. Je tiens également à le remercier pour d'autres interventions qui m'ont grandement facilité la tâche, que ce soit pour le bouclage du GRETSI ou encore la transmission de mon dernier chapitre ... Un grand merci pour tout!

J'adresse mes remerciements sincères à tous les membres du GPI pour l'aide qu'ils ont pu m'apporter, notamment aux abords de la soutenance. Merci aux anciens Thierry et Marc, à Andrea pour **PlotSlices** et ses bons tuyaux sur Florence (et bonne route à l'ATO!), Myriam pour nos nombreuses discussions tomographiques, qu'elles soient planes ou compactes, Amandine pour son gâteau au chocolat (euh non, pour sa tarte!), et aux séparateurs fous Vincent et Hicham. Merci également à tous les participants (même aux mauvais perdants) des nombreux « Blastin'time », « On place », « On lance la sauce? » ou autres « D'accord mais rapide » qui ont jalonné ces années. Cela me mène naturellement

à Kader, que je remercie pour sa bonne humeur, sa compréhension et l'efficacité de ses dépannages informatiques.

Enfin, comment pourrais-je oublier Sébastien, grand animateur des cafés par la créativité de ses illustrations ? Merci pour tout et surtout pour les conseils stratégiques qu'il a pu me donner tout au long de cette thèse. Je remercie Pierre BRÉMAUD (le *sage*) pour la qualité des sièges de sa Passat, mais surtout pour s'être abstenu de poser des questions embarrassantes le jour J ! Enfin, un grand merci à l'administration du laboratoire, et une mention spéciale à Mourad pour son soutien lors des nombreuses nocturnes qui ont jalonné ma fin de thèse.

Table des matières

Liste des notations	1
Liste des figures	5
Liste des tableaux	7
Présentation du document	9
I Introduction	13
I.1 Contexte du problème	13
I.1.1 Nature des données	13
I.1.2 Nature de la scène	15
I.1.3 Conclusions	16
I.2 Modélisation des mesures	17
I.2.1 Projections en 2D	17
I.2.2 Projections en 3D	17
I.2.3 Problème direct	19
I.3 Méthodes classiques de reconstruction	19
I.3.1 Méthodes analytiques	20
I.3.2 Méthodes algébriques	22
I.3.3 Conclusions	23
I.4 Méthodologie bayésienne	24
I.5 Vers la modélisation de la scène	25
I.6 Conclusions	26
II Approche par voxels	27
II.1 Modélisation du problème	27
II.1.1 Remarques préliminaires	27
II.1.2 Estimation de la scène	28
II.1.3 Organisation du chapitre	28
II.2 Construction du critère régularisé	28
II.2.1 Choix d'un potentiel convexe	29
II.2.2 Choix d'un potentiel non convexe	30
II.2.3 Choix du critère	31
II.3 Optimisation du critère	32
II.3.1 Principe de l'algorithme	32
II.3.2 Convergence	33
II.3.3 Choix retenu	34
II.3.4 Initialisation	35
II.4 Réduction du nombre de voxels	35
II.5 Stratégie multirésolution	37
II.5.1 Principe	37
II.5.2 Réglage des hyperparamètres	38
II.6 Conclusions	39

II.A Preuves des propositions 1 et 2	39
III Approche contour	41
III.1 Introduction	41
III.2 Modélisations de contours	42
III.2.1 Modélisations non paramétriques	42
III.2.2 Modélisations paramétriques	44
III.3 Sélection de modèles adéquats	47
III.3.1 Modèles à faible nombre de paramètres	47
III.3.2 Modèles à grand nombre de paramètres	47
III.3.3 Estimation des paramètres	48
III.4 Étude des modèles retenus	50
III.4.1 Ellipsoïdes et superquadriques	50
III.4.2 Contours harmoniques	52
III.4.3 Splines	54
III.5 Conclusions	56
IV Approche polygonale	57
IV.1 Introduction	57
IV.2 Modèle polygonal	58
IV.2.1 Méthode des moments	58
IV.2.2 Approche proposée	60
IV.3 Optimisation du critère	64
IV.3.1 Techniques locales d'optimisation	64
IV.3.2 Stratégie retenue	66
IV.3.3 Initialisation et critère d'arrêt	67
IV.3.4 Conclusions	68
IV.4 Étude des projections	68
IV.4.1 Calcul approché	69
IV.4.2 Calcul direct	69
IV.4.3 Mise à jour des projections	73
IV.4.4 Calcul du gradient du critère	74
IV.5 Conclusions	76
IV.A Caractère non convexe de $\mathcal{K}$	76
IV.A.1 Cas $r = 1$	77
IV.A.2 Cas $r > 1$	78
V Approche polyédrique	79
V.1 Introduction	79
V.2 Modèle polyédrique	79
V.2.1 Méthode des moments	80
V.2.2 Modèle proposé	80
V.2.3 Estimation des sommets au sens du MAP	81
V.3 Optimisation du critère	83
V.3.1 Techniques locales d'optimisation	83
V.3.2 Difficultés de mise en œuvre	84
V.3.3 Conclusions	84
V.4 Étude des projections	85
V.4.1 Calcul direct	85
V.4.2 Mise à jour des projections	86
V.4.3 Calcul du gradient du critère	88
V.5 Extraction de mini-polyèdres	90

V.5.1	Positionnement des sommets	90
V.5.2	Décomposition d'un polyèdre	92
V.5.3	Fermeture d'un mini-polyèdre	92
V.5.4	Algorithme d'extraction d'un mini-polyèdre	95
V.6	Conclusions	95
V.A	Preuve de la proposition 7	96
VI	Simulations	97
VI.1	Introduction	97
VI.1.1	Nature des données	97
VI.1.2	Qualité des reconstructions	99
VI.1.3	Organisation du chapitre	100
VI.2	Approche par voxels	100
VI.2.1	Stratégie monorésolution	100
VI.2.2	Stratégie multirésolution	102
VI.2.3	Conclusions	102
VI.3	Approche par contours paramétriques globaux	105
VI.3.1	Quadriques et superquadriques	105
VI.3.2	Modèles harmoniques	106
VI.4	Approche par splines d'ordre un	111
VI.4.1	Cas 2D	111
VI.4.2	Cas 3D	115
VI.5	Conclusions	120
	Conclusions	121
A	Méthode par voxels : caractère Lipschitz de $\nabla \mathcal{J}$	123
A.1	Représentation complète	123
A.2	Représentation creuse	124
B	Recovery of moments of a compact homogeneous object from tomographic projection data	127
B.1	Introduction	129
B.2	Summary of the 2D results	130
B.3	Reconstruction of the geometric moments in 3D case	132
B.3.1	Basic relations	132
B.3.2	First step	134
B.3.3	Following steps	134
B.3.4	Conclusion	136
B.4	Proposed Algorithms	136
B.4.1	2D case	136
B.4.2	3D case	137
B.5	Simulation and numerical results	137
B.6	Conclusion	138
C	Méthode polygonale : différenciation de $\mathcal{J}(v)$	143
C.1	Terme d'adéquation aux données	143
C.1.1	Cas où $p_t(v_1) \neq 0$	145
C.1.2	Cas où $p_t(v_1) = 0$	148
C.1.3	Cas particulier : $a = b$ ou $a = c$	149
C.2	Terme de régularisation	150

D	Méthode polyédrique : différenciation de $\mathcal{K}(\mathbf{V})$	151
D.1	Introduction	151
D.2	Changement de repère	151
D.3	Cas d'un tétraèdre	152
D.3.1	Tétraèdre de type (1)	154
D.3.2	Tétraèdre de type (2)	155
D.3.3	Tétraèdre de type (3)	156
D.3.4	Coût de la projection d'un tétraèdre et de sa dérivée	158
D.4	Différentiabilité de $\mathcal{K}(\mathbf{V})$	159
D.5	Évaluation de $\nabla \mathcal{K}(\mathbf{V})$	160
D.6	Cas où les directions ne sont pas orthogonales au détecteur	160
E	Angles solides d'un polyèdre	163
E.1	Angles solides d'un tétraèdre	163
E.2	Différenciation des angles solides pour un tétraèdre non singulier	164
E.3	Différenciation des angles solides pour un tétraèdre singulier	165
E.4	Angles solides et courbures d'un polyèdre	166
	Bibliographie	168

LISTE DES NOTATIONS

Remarque. Les notations définies ci-dessous sont celles que nous utilisons le plus dans ce document. La liste de notations n'est donc pas exhaustive.

NOTATIONS GÉNÉRALES

Fonction de classe $\mathcal{C}^p$	fonction p fois continûment différentiable
$\langle ., . \rangle$	produit scalaire euclidien sur $\mathbb{R}^p$
$\ . \ , d(. , .)$	norme et distance euclidienne sur $\mathbb{R}^p$
$\text{Supp}(\Omega)$	sous-espace de $\mathbb{R}^2$ ou $\mathbb{R}^3$ qui supporte l'ensemble Ω ($\Omega \subset \mathbb{R}^2$ ou $\mathbb{R}^3$)
$\text{Int}(\Omega)$	intérieur dans $\mathbb{R}^2$ ou $\mathbb{R}^3$ de l'ensemble Ω ($\Omega \subset \mathbb{R}^2$ ou $\mathbb{R}^3$)
$\vec{z} \in \mathbb{R}^2$	vecteur représenté par le nombre complexe z
$\partial Q / \partial z, z \in \mathbb{C}$	dérivée partielle « complexe », égale à $\partial Q / \partial x + i \partial Q / \partial y$, où $z = x + iy$
$\partial Q / \partial \mathbf{x}, \mathbf{x} \in \mathbb{R}^p$	dérivée partielle « vectorielle » : vecteur de $\mathbb{R}^p$ de i -ème composante $\partial Q / \partial x_i$

DONNÉES

$\mathcal{S}$	scène examinée
$\mathcal{C}^*$	contour de l'objet réel
m	nombre total de projections
$r, (r_x, r_y)$	nombre de pixels d'une radiographie (cas 2D, 3D)
M	taille totale des données. $M = m r$ ou $m r_x r_y$
$\mathbf{d}$	données (projections bruitées de $\mathcal{C}^*$), rangées vectoriellement
$\mathbf{d}_k = \{d_i^k\}$ ou $\{d_{ij}^k\}$	k -ème bloc de projection, vecteur de taille r ou $r_x r_y$
$\mathcal{A}$	opérateur de projection
S_k, S	position de la k -ème source
$t_k, \mathbf{t}_k$	position du premier pixel détecteur de la k -ème projection
$\delta t_k, \delta \mathbf{t}_k$	taille des pixels détecteurs pour la k -ème projection
RSB	rapport signal à bruit

TOMOGRAPHIE 2D

$f(x, y)$	fonction de densité
$\mu(x, y)$	fonction d'atténuation
θ	angle de projection
$\mathbf{w} = \mathbf{w}_{\theta - \pi/2}$	direction de projection
$p_\theta(t)$	projection parallèle d'angle θ
$\mathcal{L}_{t, \theta}$	rayon de projection décrit par $\langle \mathbf{w}_\theta, \mathbf{x} \rangle = t$

TOMOGRAPHIE 3D

$f(x, y, z)$	fonction de densité
$\mu(x, y, z)$	fonction d'atténuation
$\boldsymbol{\theta} = (\theta, \varphi)$	angle de projection en coordonnées sphériques
$\boldsymbol{w} = \boldsymbol{w}_{\boldsymbol{\theta}}$	direction de projection
Z	positionnement du détecteur (plan $z = Z$)
$p_{\boldsymbol{\theta}}(\boldsymbol{t})$	projection suivant $\boldsymbol{w}_{\boldsymbol{\theta}}$ en un pixel $\boldsymbol{t}$ du plan détecteur
$\mathcal{L}_{\boldsymbol{t}, \boldsymbol{\theta}}$	direction correspondante

MODÈLE PAR VOXELS

n	nombre de voxels binaires
$\boldsymbol{f} = [f_1, \dots, f_n]^t$	densités des voxels
$\mathcal{X}$	ensemble des contraintes sur $\boldsymbol{f}$
$[\cdot]^+$	opérateur de projection sur $\mathcal{X}$
$\boldsymbol{A}$	matrice de projection
$\boldsymbol{p}$	vecteur projection, de taille M : $\boldsymbol{p} = \boldsymbol{A}\boldsymbol{f}$
$\boldsymbol{p}_k$	k -ème bloc de projection, de taille r ou $r_x r_y$

CONTOURS DÉFORMABLES

$\mathcal{C}$	modèle de contour déformable (cadre général)
$\mathcal{C}(t), \mathcal{C}(t, u)$	paramétrisation continue en 2D et 3D
$\mathcal{C}(s)$	paramétrisation normale en 2D (s est l'abscisse curviligne)
$u(t), u(s) \in \mathbb{R}^2$	représentation 2D des contours fermés $\mathcal{C}(t), \mathcal{C}(s)$
P	périmètre du contour (cas 2D)
$\mathcal{C}_{\boldsymbol{x}}$	modèle paramétrique
n	nombre de paramètres (réels ou vectoriels) du contour
$\boldsymbol{x} \in \mathbb{R}^n, \mathbb{R}^{2n}$ ou $\mathbb{R}^{3n}$	paramètres du contour
$\mathcal{P}_{\boldsymbol{x}}$	approximation de $\mathcal{C}_{\boldsymbol{x}}$ par une spline d'ordre 1
$n_{\boldsymbol{x}}$	nombre de sommets de $\mathcal{P}_{\boldsymbol{x}}$
$\mathcal{K}(\boldsymbol{x})$	critère des moindres carrés sur $\mathcal{C}_{\boldsymbol{x}}$
$\tilde{\mathcal{K}}(\boldsymbol{x})$	critère des moindres carrés sur $\mathcal{P}_{\boldsymbol{x}}$ (approximation de $\mathcal{K}$)
$\boldsymbol{\xi}_k$	axes principaux d'inertie estimés de l'objet $\mathcal{C}^*$
$\boldsymbol{\xi} = (\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \boldsymbol{\xi}_3)$	système de coordonnées associé
G	point central pour les contours étoilés
$\rho(\theta)$	modélisation d'une courbe fermée en coordonnées polaires
$\rho(\boldsymbol{\theta}) = \rho(\theta, \varphi)$	modélisation d'une surface fermée en coordonnées sphériques
$\rho_l^k(\boldsymbol{\theta}), k \leq l$	k -ème harmonique sphérique de degré l
L	ordre du développement en série harmonique
$B_i(t)$	B-splines cubiques
P_i, P_{ij}	points de contrôle pour les modèles splines
n	nombre de points de contrôle (modèles spline)
$\mathcal{H}(\mathcal{C}_1, \mathcal{C}_2)$	distance de Hausdorff entre contours

MODÈLE POLYGONAL

$\mathcal{P}$	polygone qui représente l'objet compact
n	nombre de sommets de $\mathcal{P}$
$\mathbb{A}_n$	ensemble des polygones admissibles à n sommets
$\mathbf{v} \in \mathbb{C}^n$	sommets de $\mathcal{P}$ ordonnés dans le sens trigonométrique
$v_j = x_j + iy_j \in \mathbb{C}$	j -ème sommet
$v_j^* \in \mathbb{C}$	barycentre des voisins de v_j : $v_j^* = (v_{j-1} + v_{j+1})/2$
$A_j(\mathbf{v}) \in (0, \pi)$	angle formé par les deux arêtes adjacentes à v_j
$\kappa_j(\mathbf{v})$	courbure en v_j
$\mathbf{t}_j$	triangle (v_{j-1}, v_j, v_{j+1})
$\mathbf{f}_j$	normale à l'arête (v_j, v_{j+1}) orientée vers l'extérieur de $\mathcal{P}$
d_j	distance de cette arête à la source S
$p_{\mathbf{v}}(t)$	une projection du polygone $\mathcal{P} \equiv \mathbf{v}$ (projection d'angle θ)
$p_{\mathbf{v}}^k(t)$	k -ème projection de $\mathcal{P}$
$p_{\mathbf{v}}(v_j)$	valeur de $p_{\mathbf{v}}(t)$ en le projeté $t = \langle \vec{v}_j, \mathbf{w}_\theta \rangle$ de v_j (abus de notation)

MODÈLE POLYÉDRIQUE

$\mathcal{P}$	polyèdre qui représente l'objet compact
n	nombre de sommets de $\mathcal{P}$
$\mathbb{A}_n$	ensemble des polyèdres admissibles à n sommets
$\mathbf{V} \in \mathbb{R}^{3n}$	sommets du polyèdre
$V_i \in \mathbb{R}^3$	i -ème sommet
n_i	nombre de sommets voisins de V_i
∂i	ensemble des indices des sommets voisins de V_i
$V_{i \rightarrow k}, 1 \leq k \leq n_i$	sommets voisins de V_i
$\boldsymbol{\tau}_{ik}, 1 \leq k \leq n_i$	vecteur directeur (unitaire) de la k -ème arête adjacente à V_i , <i>i.e.</i> , $(V_i, V_{i \rightarrow k})$
$i \sim j, i \not\sim j$	relation de voisinage/non voisinage entre sommets
$V_i^* \in \mathbb{R}^3$	barycentre des sommets voisins de V_i
$A_i(\mathbf{V}) \in (0, 4\pi)$	angle solide en V_i
$\Gamma_i(\mathcal{P})$	cône correspondant
$\kappa_i(\mathbf{V})$	courbure en V_i
$\mathcal{P}_i$	mini-polyèdre formé à partir de V_i et de ses voisins
$\mathcal{Z}_i$	base de ce mini-polyèdre (polygone de $\mathbb{R}^3$)
$W_i \in \mathbb{R}^3$	point fictif qui referme $\mathcal{P}_i$
$\mathcal{T}_{ik}, 1 \leq k \leq n_i$	tétraèdres formant le mini-polyèdre $\mathcal{P}_i$
$\mathbf{n}_i$	normale sortante « approchée » en le sommet V_i
$\mathcal{F}$	facettes du polyèdre
q	nombre de facettes
$\mathcal{F}_j$	j -ème facette
$\text{Supp}(\mathcal{F}_j)$	plan support de cette facette
V_{j1}, V_{j2}, V_{j3}	sommets de $\mathcal{F}_j$
$\mathbf{f}_j$	normale à $\mathcal{F}_j$ orientée vers l'extérieur
d_j	distance algébrique de $\text{Supp}(\mathcal{F}_j)$ à la source S

$\mathcal{V}(\mathcal{P})$	volume du polyèdre
$p_{\mathcal{P}}(\mathbf{t}) \equiv p_{\mathbf{V}}(\mathbf{t})$	une projection du polyèdre (projection d'angle $\boldsymbol{\theta}$)
$p_{\mathbf{V}}^k(\mathbf{t})$	k -ème projection du polyèdre
$v_i \in \mathbb{C}$	projeté de V_i sur le plan détecteur
$p_{\mathbf{V}}(V_i) = p_{\mathbf{V}}(v_i)$	valeur de $p_{\mathbf{V}}(\mathbf{t})$ en le sommet projeté $\mathbf{t} = \vec{v}_i$ (abus de notation)
$\delta r_k(i)$	nombre de pixels détecteurs inclus dans le support de $p_{\mathcal{P}_i}^k(\mathbf{t})$

ÉNERGIES

$\mathcal{J}$	énergie <i>a posteriori</i>
$\nabla \mathcal{J}$	gradient de $\mathcal{J}$
$\mathcal{K}$	terme d'adéquation aux données
$\mathcal{R}$	énergie de régularisation (forme générale)
$\mathcal{D}(\mathbf{f})$	approche par voxels : énergie de régularisation markovienne
λ	paramètre de régularisation associé à $\mathcal{D}(\mathbf{f})$ ou $\mathcal{R}(\mathcal{C})$
ϕ_T	pénalisation 1D relative à $\mathcal{D}(\mathbf{f})$
$\mathcal{U}(\mathbf{f})$	approche par voxels : énergie de rappel à 0
μ	paramètre de régularisation associé à $\mathcal{U}(\mathbf{f})$

ALGORITHMES

K	nombre d'itérations
$\mathcal{P}^{(0)}, \mathbf{V}^{(0)}, \mathbf{f}^{(0)}$	estimée initiale
$\mathcal{P}^{(k)}, \mathbf{V}^{(k)}, \mathbf{f}^{(k)}$	estimée à l'itération k . En particulier, $\mathbf{V}^{(k)} = \{V_1^{(k)}, \dots, V_n^{(k)}\}$
$\mathbf{a}_k$	direction de descente à l'itération k (algorithme de gradient)
s_k	pas de descente direction à l'itération k (algorithme de gradient)
$\hat{\mathcal{P}}, \hat{\mathbf{V}}, \hat{\mathbf{f}}$	estimée finale, après K itérations
$\mathbf{f}^*$	maximum <i>a posteriori</i> (approche par voxels)

MULTIRÉSOLUTION (APPROCHE PAR VOXELS)

R	nombre de niveaux de résolution
$\mathcal{J}_r$	énergie <i>a posteriori</i> à la résolution r
$\mathcal{I}_r$	transformation de l'image lors du r -ème incrément de résolution
$\mathbf{f}_r^{(0)}$	initialisation à la résolution r
$\hat{\mathbf{f}}_r$	estimée à la résolution r
v_r	nombre de voxels actifs à la résolution r
$\mathcal{V}_r$	ensemble des voxels actifs à la résolution r

Liste des Figures

I.1	Mesure d'une projection en tomographie à rayons X.	14
I.2	Présentation du problème.	15
I.3	Mode de projection orthogonale en 2D.	16
I.4	Mode de projection orthogonale en 3D.	18
I.5	Représentation des données dans le plan de Fourier.	20
I.6	Projection d'une scène 2D pixellisée.	22
II.1	Choix classiques de potentiels convexes.	29
II.2	Principe du GNC.	31
II.3	Illustration de la méthode du gradient projeté.	33
II.4	Division d'un voxel actif en 8 « sous-voxels » identiques de même densité.	36
II.5	Schéma de la stratégie multirésolution.	37
III.1	Interprétation du modèle harmonique 2D par une superposition d'ellipses.	46
III.2	Modélisations 2,5D.	48
III.3	Mécanisme de projection approchée d'un contour paramétré.	49
III.4	Paramétrisation d'un ellipsoïde.	50
III.5	Paramétrisation d'un super-ellipsoïde.	51
IV.1	Exemple d'auto-intersection d'un polygone.	58
IV.2	Paramétrisation normale d'une courbe régulière.	61
IV.3	Paramétrisation d'un polygone, et notions relatives à sa courbure.	62
IV.4	Régularisation par la douceur locale du polygone.	63
IV.5	Évaluation <i>approchée</i> d'une projection d'un polygone.	68
IV.6	Évaluation <i>exacte</i> d'une projection d'un polygone dans le cas régulier.	70
IV.7	Évaluation <i>exacte</i> d'une projection d'un polygone dans le cas singulier.	70
IV.8	Statut des sommets d'un polygone.	73
IV.9	Structure des projections d'un triangle.	77
V.1	Angles solides d'un tétraèdre.	82
V.2	Intersection d'un rayon et d'une facette triangulaire.	85
V.3	Extraction d'un mini-polyèdre.	87
V.4	Effet d'un sommet sur la projection d'un polyèdre.	87
V.5	Positionnement d'un sommet d'un polyèdre.	91
V.6	Fermeture d'un mini-polyèdre dont la base est régulière.	92
V.7	Triangulation de la base par le rajout du point fictif V_1^*	93
V.8	Fermeture de la base par le rajout du point fictif W_1 de $(V_1 V_1^*)$	94
V.9	Cas particulier où V_1 et V_1^* sont proches.	94
VI.1	Données synthétiques 2D.	98
VI.2	Données synthétiques 3D.	98
VI.3	Reconstruction par voxels en monorésolution.	101
VI.4	Reconstruction par voxels en multirésolution.	103
VI.5	Convergences de l'algorithme du gradient projeté.	103

VI.6	Quantification des reconstructions par voxels en monorésolution et multirésolution. . .	104
VI.7	Reconstructions de la surface (IV) par un ellipsoïde et une superquadrique.	105
VI.8	Reconstructions harmoniques du contour (II).	107
VI.9	Reconstructions elliptiques et harmoniques du contour (I).	107
VI.10	Reconstructions harmoniques de la surface (III).	108
VI.11	Reconstructions harmoniques de la surface (IV).	109
VI.12	Influence du nombre de paramètres du développement harmonique sur la reconstruction de la surface (III).	110
VI.13	Influence de l'approximation polyédrique du modèle harmonique sur la reconstruction de la surface (III).	110
VI.14	Reconstruction polygonale du contour (I) par moindres carrés.	111
VI.15	Reconstruction polygonale du contour (I) par l'approche régularisée.	112
VI.16	Évaluation de la qualité de la reconstruction polygonale par l'approche régularisée. .	113
VI.17	Influence de l'initialisation sur la reconstruction polygonale.	114
VI.18	Convergence de l'algorithme GS (en 2D) pour la minimisation du critère pénalisé. . .	115
VI.19	Choix de l'algorithme d'optimisation locale pour la méthode polyédrique.	115
VI.20	Reconstructions polyédriques de la surface (III).	116
VI.21	Reconstructions polyédriques de la surface (IV) avec une initialisation ellipsoïdale. .	117
VI.22	Influence de l'hyperparamètre λ sur les reconstructions polyédriques.	118
VI.23	Reconstructions polyédriques de la surface (IV) avec une superquadrique pour initialisation.	118
VI.24	Reconstructions polyédriques de la surface (IV) avec des initialisations harmoniques. .	119
VI.25	Influence de l'hyperparamètre λ sur la reconstruction polyédrique associée à une initialisation harmonique particulière.	119
B.1	Geometric definition of a projection.	132
B.2	The polyhedral object and the ellipsoid.	138
B.3	Noiseless and noisy projections of the polyhedron.	140
B.4	Noiseless and noisy projections of the ellipsoid.	141
C.1	Projection d'un triangle : influence d'un sommet « de projection non nulle ».	144
C.2	Déplacement d'un sommet « de projection non nulle » dans la direction x	144
C.3	Autres déplacements d'un sommet « de projection non nulle ».	146
C.4	Projection d'un triangle : influence d'un sommet « de projection nulle ». Cas (2-a) et (2-b).	147
C.5	Évaluation de la valeur maximale de la projection d'un triangle (cas (2-a)).	148
C.6	Évaluation de la valeur maximale de la projection d'un triangle (cas (2-b)).	149
C.7	Courbure d'un triangle.	150
D.1	Support de la projection d'un tétraèdre.	153
D.2	Représentation de la projection d'un tétraèdre.	153
D.3	Déplacement du sommet dans la direction x	154
D.4	Contribution d'un tétraèdre aux dérivées partielles de $\mathcal{K}(\mathbf{V})$	159
E.1	Angle solide d'un tétraèdre plan.	165

Liste des Tableaux

II.1	Algorithme du gradient projeté pour la reconstruction par voxels.	34
II.2	Algorithme multirésolution pour la reconstruction par voxels.	36
IV.1	Méthode locale de descente par blocs (sommet par sommet).	65
IV.2	Test d'acceptabilité pour la procédure de descente par blocs.	66
IV.3	Algorithme de projection d'un polygone convexe.	71
IV.4	Algorithme de projection d'un polygone non convexe.	72
V.1	Mise à jour des projections d'un polyèdre suite au déplacement d'un sommet.	88
V.2	Génération d'un mini-polyèdre.	95
VI.1	Données 3D. Description des vecteurs directeurs des projections.	99
VI.2	Qualité des reconstructions polyédriques de la surface (IV).	120
B.1	Polyhedron - Reconstruction of the moments up to the second order.	138
B.2	Ellipsoid - Reconstruction of the moments up to the second order.	139
B.3	Ellipsoid - Reconstruction of the third order moments.	139
D.1	Algorithme de parcours d'un triangle pour la différenciation de $\mathcal{K}(\mathbf{V})$	160
E.1	Procédure de calcul du terme de régularisation et de son gradient.	167

PRÉSENTATION DU DOCUMENT

Cette thèse concerne la reconstruction 3D d'images binaires en tomographie X à partir d'un faible nombre de vues. En contrôle non destructif, une première expertise peut révéler la présence d'*une seule* zone défectueuse, ou occlusion (trou d'air) dans un milieu homogène (l'acier). L'image reconstruite correspond donc à la carte de densités de la zone inspectée, zone contenant le défaut. Ce dernier est alors considéré comme un *objet* compact et homogène, et deux approches sont envisageables.

1. Modéliser l'image par un champ binaire (0 pour l'acier et 1 pour l'air) et estimer les valeurs des voxels de l'image à partir des données (Dinten, 1990; Sauer et al., 1994; Gautier, 1996).
2. Modéliser directement le *contour* du défaut (Hanson et al., 1994; Milanfar et al., 1994).

La première, plus classique, présente l'avantage de mener à un problème direct linéaire, ce qui simplifie nettement la reconstruction. Pour la seconde, le problème direct n'est pas linéaire, mais le nombre de paramètres à estimer est singulièrement réduit (quelques centaines contre quelques millions pour la première approche). Nous avons choisi de développer particulièrement la seconde approche, qui de plus, prend en compte explicitement la compacité de l'objet.

Les deux problèmes de reconstruction sont rendus très mal posés par le fait que les projections sont en nombre limité et à angles limités. Il est alors indispensable d'apporter de l'information *a priori* sur l'objet et/ou le fond pour régulariser le problème. Une approche possible est la modélisation probabiliste de cette information et l'estimation bayésienne (Mohammad-Djafari et Soussen, 1999).

Organisation du manuscrit

Le manuscrit est composé de 6 chapitres et s'organise comme suit :

- Le chapitre I positionne le problème et décrit les méthodes de reconstruction adaptées.
- Le chapitre II synthétise les méthodes par voxels.
- Le chapitre III est une introduction aux méthodes par contours.
- Les chapitres IV et V développent les modèles polygonal et polyédrique pour la reconstruction de contours fermés 2D et 3D.
- Le chapitre VI illustre et compare les performances des méthodes décrites précédemment.

Introduction

Au chapitre I, nous décrivons le contexte des problèmes de reconstruction d'objets compacts binaires 2D et 3D, et nous justifions l'échec des méthodes «classiques» de reconstruction (inversion généralisée, synthèse de Fourier, rétroprojection filtrée) en insistant sur le caractère mal conditionné du problème. Pour obtenir des solutions acceptables, nous avons recours aux techniques de régularisation, et utilisons la méthodologie bayésienne. Enfin, nous décrivons les grandes classes de modèles de la scène : les modèles par voxels (développés au chapitre II), et les modèles par contours (développés aux chapitres III, IV et V). Nous justifions le choix d'étudier plus particulièrement ces derniers.

Modèles par voxels

Dans le chapitre II, nous traitons les modèles par voxels, qui sont les plus utilisés en reconstruction tomographique. La première partie de ce chapitre réalise une synthèse des méthodes de reconstruction associées à ces modèles, qui estiment généralement une solution régularisée. Par exemple, on envisage l'estimation de la scène au sens du *maximum a posteriori* (MAP), en minimisant un critère composite. Par la suite, nous associons une stratégie par voxels à notre problème afin d'effectuer ultérieurement des comparaisons avec les méthodes par contour. Par simplicité, nous définissons un critère composite *convexe*. Nous envisageons les difficultés algorithmiques liées à son optimisation. En particulier, nous développons une stratégie multirésolution originale afin de réduire le coût numérique de la reconstruction.

Modèles par contour

Le chapitre III réalise une présentation des modèles par contours pour la reconstruction d'objets compacts 2D et 3D. Dans la littérature du traitement d'images, ces modèles ont davantage été utilisés pour la segmentation et la vision par ordinateur. La première partie du chapitre réalise une classification de ces modèles, en soulignant les modèles adaptés à notre contexte. Dans la seconde partie, nous sélectionnons des modèles *paramétriques* adéquats, en insistant sur leurs caractéristiques et leurs limites. Après avoir proposé une structure d'estimation de ces contours (*i.e.*, de leurs paramètres), nous développons les modèles sélectionnés. En particulier, nous mettons en valeur les modèles *locaux* (dont chaque paramètre ne contrôle qu'une portion spécifique du contour) pour la reconstruction de formes complexes, et sélectionnons les modèles par splines. Nous discutons l'influence de l'ordre de la spline, et optons pour les splines d'ordre un. Ce dernier modèle est développé plus particulièrement aux chapitres IV et V dans les cas 2D et 3D.

Modèle par spline d'ordre un

Les chapitres IV et V représentent la partie principale de cette thèse. Bien que les principes des reconstructions 2D et 3D soient similaires, l'exposé du cas 2D apparaît plus simple. Par conséquent, nous choisissons de développer le modèle polygonal au chapitre IV, puis de l'étendre au cas 3D dans le chapitre V. Cette extension nécessite un investissement algorithmique important, qui constitue la base de notre contribution.

Au chapitre IV, nous définissons le modèle polygonal $\mathcal{P} = \{v_1, \dots, v_n\}$ ($v_i \in \mathbb{C}$), et proposons une approche régularisée pour sa reconstruction. Elle consiste à estimer les positions $\hat{v}_i$ des sommets au sens du MAP. La régularisation fait intervenir des informations *a priori* sur la douceur locale du contour. Nous envisageons plusieurs termes de douceur, en sélectionnant celui qui apparaît le plus adapté. Nous montrons que le problème d'optimisation qui en résulte est complexe, car l'énergie *a posteriori* $\mathcal{J}(v_1, \dots, v_n)$ est généralement non convexe et multimodale. Dans la suite de ce chapitre, nous étudions cette optimisation, et nous optons pour des stratégies déterministes itératives d'optimisation locale, dont le résultat dépend fortement de l'initialisation. Nous montrons l'intérêt d'un algorithme de descente par blocs (où les blocs sont les sommets $v_i \in \mathbb{C}$), qui ré-estime successivement (et cycliquement) les positions $\hat{v}_i$ des sommets de $\mathcal{P}$. Nous montrons en particulier que cet algorithme permet de tester que le polygone reconstruit est valide, c'est-à-dire qu'il ne s'intersecte pas. Dans la dernière partie du chapitre, nous étudions les difficultés de mise en œuvre liées à l'optimisation locale de $\mathcal{J}$. Nous présentons en détail les éléments essentiels à la manipulation des polygones : le calcul de leurs projections, et la remise à jour de celles-ci suite au déplacement de l'un des sommets de $\mathcal{P}$. En particulier, ce dernier point est la base de notre procédure originale de test de validité des polygones. Les parties les plus techniques sont présentées en annexe C.

Le chapitre V concerne la reconstruction 3D proprement dite, qui constitue l'objectif du travail de thèse. Il est bâti de la même façon que le chapitre IV, et nous insistons peu sur les parties communes aux modèles 2D et 3D. Par contre, nous soulignons leurs différences, et les difficultés inhérentes à l'aspect tridimensionnel du problème. La première différence concerne la définition du modèle. Un polyèdre n'est pas simplement défini par la séquence $\mathbf{V} = \{V_1, \dots, V_n\}$ ($V_i \in \mathbb{R}^3$) de ses sommets, mais aussi par les facettes qui le constituent. Par simplicité, nous optons pour des facettes triangulaires, que nous fixons au préalable. Nous ramenons comme dans le cas 2D la reconstruction de $\mathcal{P}$ à l'estimation des positions $\hat{V}_i$ de ses sommets. Nous réalisons cette estimation au sens du MAP, en incorporant des informations *a priori* de douceur du contour. L'originalité de notre contribution ne concerne pas la méthodologie adoptée, mais les choix de la régularisation et la mise en œuvre de l'optimisation de l'énergie *a posteriori*. Elle est menée par un algorithme itératif local, mais son choix s'avère moins simple que dans le cas 2D. En effet, le temps de calcul devient un paramètre fondamental, et la méthode de descente par blocs (sommets par sommets) s'avère très lente pour les modèles à grand nombre de sommets. Dans ces cas, nous préférons une méthode de descente globale (directement sur $\mathbf{V} \in \mathbb{R}^{3n}$). Comme dans le cas 2D, nous concevons une procédure originale de calcul du gradient des projections de $\mathcal{P}$, qui s'appuie sur la mise à jour de ses projections suite au déplacement de l'un des sommets V_i et sur l'extraction et la «tétraédrisation» d'un mini-polyèdre local au sommet V_i . Les parties techniques de ce chapitre reposent largement sur l'annexe D.

Illustration des méthodes

Enfin, le chapitre VI illustre sur des jeux données synthétiques les méthodes de reconstruction présentées dans les chapitres précédents. Après avoir présenté des reconstructions par voxels (en monorésolution puis en multirésolution), nous montrons l'apport du modèle par contour. Nous soulignons les intérêts et les limitations des différents modèles paramétriques introduits au chapitre III, et nous quantifions en particulier l'apport des modèles par splines d'ordre un. Nous justifions leur capacité à reconstruire en un temps raisonnable des formes complexes, puis discutons l'influence de l'initialisation, du nombre de sommets et du paramètre de régularisation sur la surface reconstruite.

CHAPITRE I

INTRODUCTION

- I.1 Contexte du problème
- I.2 Modélisation des mesures
- I.3 Méthodes classiques de reconstruction
- I.4 Méthodologie bayésienne
- I.5 Vers la modélisation de la scène
- I.6 Conclusions

I.1 Contexte du problème

Notre étude concerne la reconstruction d'objets compacts et homogènes à partir d'un faible nombre de projections. Ce problème se trouve au cœur d'un certain nombre de techniques d'imagerie en contrôle non destructif (CND) des matériaux, où l'on cherche à déterminer la forme précise d'un trou à l'intérieur d'un objet métallique. Les techniques utilisées à cet effet sont les rayons X (radiographie), les ultrasons, l'imagerie microonde et les courants de Foucault. Dans leurs modalités actuelles, on se contente souvent de détecter simplement la présence d'un défaut. Mais de plus en plus, le besoin de mieux caractériser cette inhomogénéité se présente. On distingue alors deux catégories de méthodes :

- Celles qui utilisent les techniques d'intelligence artificielle pour identifier un défaut parmi les défauts catalogués dans un « dictionnaire ».
- Celles qui cherchent directement à reconstruire une image de l'intérieur de l'objet, à partir de laquelle on effectue des mesures sur la forme du défaut.

Dans certains cas, la forme du défaut est complexe et les premières techniques ne donnent pas satisfaction. L'expertise requiert alors la reconstruction de la pièce. Dans ce travail, nous considérons la *tomographie à rayons X*, et nous reconstruisons des images représentant des cartes de densités des matériaux inspectés à partir des données radiographiques. Dans les paragraphes suivants, nous précisons la nature des données et leur configuration, puis nous formalisons des hypothèses quant à l'intérieur de la scène examinée.

I.1.1 Nature des données

Les données, que nous appellerons « projections » par abus de langage, sont constituées de m images 2D (ou radiographies) définies par $r_x \times r_y$ pixels sous forme de niveaux de gris. Le processus de mesure se modélise par une loi exponentielle de type :

$$I_D = I_0 \exp \left(- \int_{\mathcal{L}} \mu(x, y, z) f(x, y, z) dl \right), \quad (\text{I.1})$$

où :

- $\mathcal{L}$ désigne le support d'un rayon de projection,
- I_0 désigne la valeur de l'intensité de la source émettrice,
- I_D est la valeur mesurée,

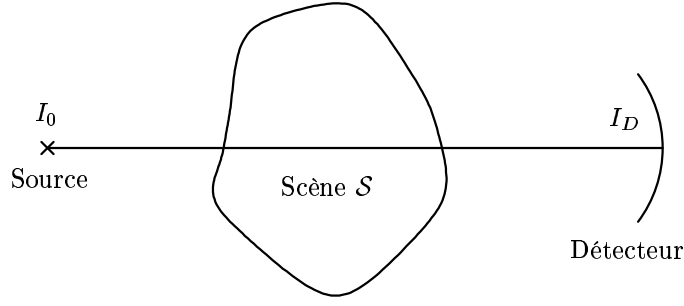


Figure I.1 – Mesure d’une projection en tomographie à rayons X.

- la fonction f modélise la densité de la scène inspectée $\mathcal{S}$,
- la fonction μ mesure les coefficients d’atténuation des matériaux de la scène.

Ce modèle est illustré par la figure I.1. Il est simple car il ne tient pas compte des imperfections des éléments radiographiques : le caractère non ponctuel des sources et du détecteur, la présence d’un flux diffusé, et la non linéarité de la réponse du détecteur. La valeur de I_0 étant connue, nous pouvons supposer, quitte à effectuer une transformation des données ($-\log I_D/I_0$), que les données correspondent aux valeurs :

$$\int_{\mathcal{L}} \mu(x, y, z) f(x, y, z) dl. \quad (\text{I.2})$$

Les données transformées sont donc les sommes pondérées des épaisseurs des matériaux traversés.

En CND, on dispose souvent d’une bonne connaissance des matériaux de la scène. L’hypothèse de l’homogénéité des matériaux implique la détermination explicite de la fonction $\mu(x, y, z)$, et la reconstruction de la scène $\mathcal{S}$ est alors équivalente à l’estimation de la densité $f(x, y, z)$ sur $\mathcal{S}$. Ce raisonnement n’est pas valable pour toutes les applications, notamment dans le domaine médical, où la scène présente souvent des zones inhomogènes. Sa reconstruction consiste alors en l’estimation conjointe des fonctions $f(x, y, z)$ et $\mu(x, y, z)$. Dans ce manuscrit, nous supposons la fonction d’atténuation entièrement déterminée. Le problème de l’estimation de $f(x, y, z)$ est tout de même extrêmement difficile pour deux raisons :

- Les données sont en nombre limité ($m \leq 10$) en raison du processus d’acquisition des radiographies : la durée d’une acquisition (pour une radiographie) se chiffre entre 7 et 10 heures.
- De plus, les angles des projections sont limités, c’est-à-dire que l’on ne peut tourner à souhait autour de la scène à cause de contraintes physiques imposées par l’environnement de l’objet. En particulier, le plan détecteur est le même pour toutes les projections.

Sur la figure I.2, nous représentons une configuration typique de mesures : les sources et le plan détecteur sont disposés de part et d’autre de la scène $\mathcal{S}$, et les données englobent totalement le support des projections de la scène (et *a fortiori* du défaut, totalement inclus dans $\mathcal{S}$). Nous projetons suivant la verticale ascendante, et le détecteur est placé dans le plan $z = Z > 0$. Les sources sont disposées dans le plan $z = 0$, sur ou au centre d’un cercle dont le rayon est de l’ordre de grandeur de la scène parallélépipédique. En pratique, les projections sont coniques, mais nous supposons que les sources sont suffisamment éloignées du détecteur pour effectuer l’approximation parallèle. Toutefois, une grande partie des méthodes que nous exposerons dans ce manuscrit sont adaptables au cas de projections coniques.

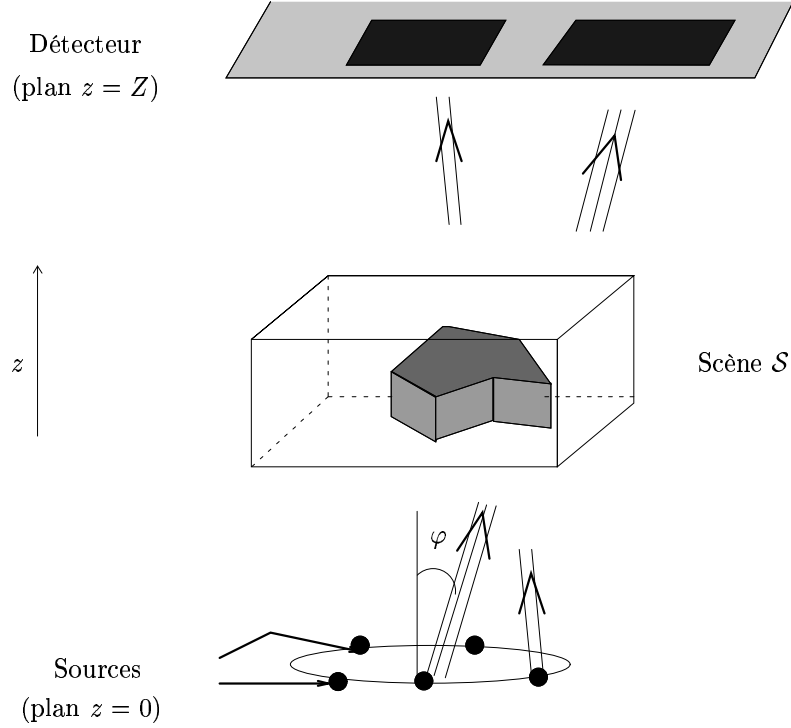


Figure I.2 – Présentation du problème. Les emplacements des radiographies englobent le support des projections du défaut.

I.1.2 Nature de la scène

La scène en elle-même est supposée contenir deux types de matériaux *homogènes* de densités respectives f_a et f_b : l'air et l'acier. La fonction de densité $f(x, y, z)$ s'exprime donc sous la forme :

$$f(x, y, z) = \begin{cases} f_a & \text{si } (x, y, z) \text{ se trouve à l'intérieur du défaut,} \\ f_b & \text{si } (x, y, z) \text{ se trouve dans l'acier.} \end{cases} \quad (\text{I.3})$$

De la même façon, la fonction d'atténuation prend alternativement les valeurs μ_a et μ_b . Faisons l'hypothèse qu'une expertise ait été réalisée au préalable et indique la présence d'*un seul* défaut *compact* (trou d'air) dans la matière. Les données (I.2) se transforment donc en :

$$f_a \mu_a \int_{\mathcal{L} \cap \text{air}} dl + f_b \mu_b \int_{\mathcal{L} \cap \text{acier}} dl. \quad (\text{I.4})$$

Dans toute la suite, nous appelons l'*objet* le volume de la zone défectueuse remplie d'air et le *fond* le reste de la scène.

Simplification du modèle. Nous ramenons à présent la scène à une image *binaire*. Plus précisément, nous nous ramenons au cas où la fonction d'atténuation n'intervient pas ($\mu_a = \mu_b = 1$), et les densités sont $f_a = 1$ et $f_b = 0$. Pour cela, considérons la scène virtuelle $\mathcal{S}_{10}$ (resp. $\mathcal{S}_{11}$) obtenue à partir de $\mathcal{S}$ en remplaçant les valeurs de μ_a, μ_b, f_a, f_b par 1,1,1,0 (resp. 1,1,1,1, la scène virtuelle $\mathcal{S}_{11}$ est donc uniforme). En notant $\mathcal{A}$ l'opérateur de projection (qui correspond aux transformations (I.4)), un calcul simple qui s'appuie sur la linéarité de cet opérateur montre que

$$\mathcal{A}(\mathcal{S}) = (f_a \mu_a - f_b \mu_b) \mathcal{A}(\mathcal{S}_{10}) + f_b \mu_b \mathcal{A}(\mathcal{S}_{11}).$$

Comme la quantité $\mathcal{A}(\mathcal{S}_{11})$ est la projection d'une scène uniforme, elle ne dépend que du mode d'acquisition des projections (c'est-à-dire des angles de projections et de la position de la scène relativement au dispositif de projection). Par conséquent, μ_a, μ_b, f_a, f_b étant connus, il est équivalent d'estimer $\mathcal{S}$ et $\mathcal{S}_{10}$ à partir des données. Dans toute la suite, nous supposons donc que :

$$\mu_a = \mu_b = 1, \quad f_a = 1 \quad \text{et} \quad f_b = 0. \quad (\text{I.5})$$

La fonction de densité $f(x, y, z)$ est donc redéfinie par :

$$f(x, y, z) = \begin{cases} 1 & \text{si } (x, y, z) \text{ se trouve à l'intérieur du défaut,} \\ 0 & \text{si } (x, y, z) \text{ se trouve dans l'acier.} \end{cases} \quad (\text{I.6})$$

I.1.3 Conclusions

En vertu des développements précédents, le problème de reconstruction se ramène à l'estimation de la densité d'une scène $\mathcal{S}$ *binaire* à partir d'un faible nombre de ses projections. Ce problème est rendu très complexe par le déficit d'information des données. En effet, le fait que les angles soient limités, c'est-à-dire que l'on ne puisse pas tourner uniformément autour de l'objet (contrairement à la majorité des applications du domaine médical) appauvrit grandement les informations disponibles sur la scène. Nous reviendrons sur ce point dans la description des méthodes de reconstruction (§ I.3).

Dans ce manuscrit, nous procéderons généralement en deux temps. Nous commencerons par envisager le problème plus simple de la reconstruction d'objets compacts binaires 2D à partir d'un faible nombre de leurs projections 1D, avant de le généraliser au cas 3D, qui constitue notre motivation *in fine*. La suite du présent chapitre est organisée comme suit. Nous commençons par formaliser les problèmes 2D et 3D (§ I.2), puis nous étudions les méthodes de reconstruction adaptées à notre contexte (§ I.3 et I.4). Enfin, nous sélectionnons des classes de modèles pertinents pour la scène (§ I.5).

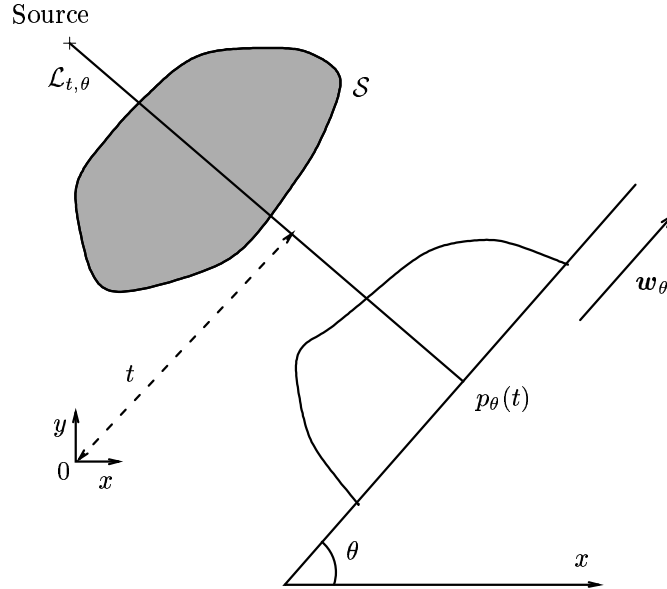


Figure I.3 – Mode de projection orthogonale en 2D.

I.2 Modélisation des mesures

Dans cette partie, nous donnons une définition mathématique de l'opérateur de projection déjà introduit ci-dessus. Il s'agit de la transformée de Radon en 2D et de la transformée à rayons X en 3D. Ces développements aboutissent à une formalisation du problème direct (§ I.2.3), pour laquelle nous préciserons notamment la nature des approximations effectuées.

I.2.1 Projections en 2D

Nous définissons les projections de la scène $\mathcal{S}$ par sa *transformée de Radon*, c'est-à-dire ses intégrales linéiques. Pour cela, nous fixons un référentiel (O, x, y) et définissons un système de coordonnées polaires (t, θ) . Toute droite $\mathcal{L}$ du plan est donc définie par un couple $(t, \theta) \in \mathbb{R}_+ \times (0, 2\pi)$:

$$\forall M = (x, y)^t \in \mathcal{L}, \quad x \cos \theta + y \sin \theta = t, \quad (\text{I.7})$$

ou vectoriellement :

$$\forall M = (x, y)^t \in \mathcal{L}, \quad \langle \mathbf{OM}, \mathbf{w}_\theta \rangle = t, \quad (\text{I.8})$$

où $\mathbf{w}_\theta = (\cos \theta, \sin \theta)^t$ est un vecteur unitaire orthogonal à $\mathcal{L}$ (voir la figure I.3). Avec ces notations, l'intégrale linéique de la scène $\mathcal{S}$ suivant $\mathcal{L} = \mathcal{L}_{t,\theta}$ est définie par :

$$p_\theta(t) = \int_{\mathcal{L}_{t,\theta}} \mu(x, y) f(x, y) dl \quad (\text{I.9})$$

$$= \iint_{\mathbb{R}^2} \mu(x, y) f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy, \quad (\text{I.10})$$

où δ désigne la distribution de Dirac 1D centrée. La transformation (I.10) est également appelée *transformée de Radon* de l'image f .

En pratique, la k -ème *projection* de la scène ($1 \leq k \leq m$) est constituée des r valeurs correspondant à l'échantillonnage du signal $p_{\theta_k}(t)$:

$$\forall i \in \{0, \dots, r-1\}, \quad p_{i+1}^k = p_{\theta_k}(t_k + i \delta t_k), \quad (\text{I.11})$$

où t_k est la position du centre du premier pixel et δt_k est le pas d'échantillonnage.

I.2.2 Projections en 3D

De même que précédemment, nous exprimons les intégrales linéiques d'une scène 3D. Nous considérons la projection suivant la direction $\mathbf{w}_\theta$ sur un plan $\mathbf{w}_\theta^\perp$ orthogonal à cette direction¹. Le vecteur $\mathbf{w}_\theta$ est défini en coordonnées sphériques et dépend d'un couple d'angles $\boldsymbol{\theta} = (\theta, \varphi) \in (0, 2\pi) \times (0, \pi)$:

$$\mathbf{w}_\theta = [\sin \varphi \cos \theta, \sin \varphi \sin \theta, \cos \varphi]^t, \quad (\text{I.12})$$

(voir la figure I.4). Nous définissons deux vecteurs $\mathbf{u}_\theta$ et $\mathbf{v}_\theta$ tels que $(\mathbf{u}_\theta, \mathbf{v}_\theta, \mathbf{w}_\theta)$ forme un repère orthonormal direct ($\mathbf{u}_\theta$ et $\mathbf{v}_\theta$ génèrent alors le plan $\mathbf{w}_\theta^\perp$) :

$$\mathbf{u}_\theta = [-\cos \varphi \cos \theta, -\cos \varphi \sin \theta, \sin \varphi]^t \quad \text{et} \quad \mathbf{v}_\theta = [\sin \theta, -\cos \theta, 0]^t. \quad (\text{I.13})$$

¹En pratique, les projections sont effectuées sur un détecteur fixe (voir la figure I.2), mais pour une plus grande clarté de l'exposé, nous choisissons d'exprimer les projections sur les plans $\mathbf{w}_\theta^\perp$. En fait, nous pouvons montrer qu'il est possible, par des transformations géométriques, d'« associer » un jeu de projections sur des détecteurs quelconques à un jeu de projections sur des plans orthogonaux aux directions de projection.

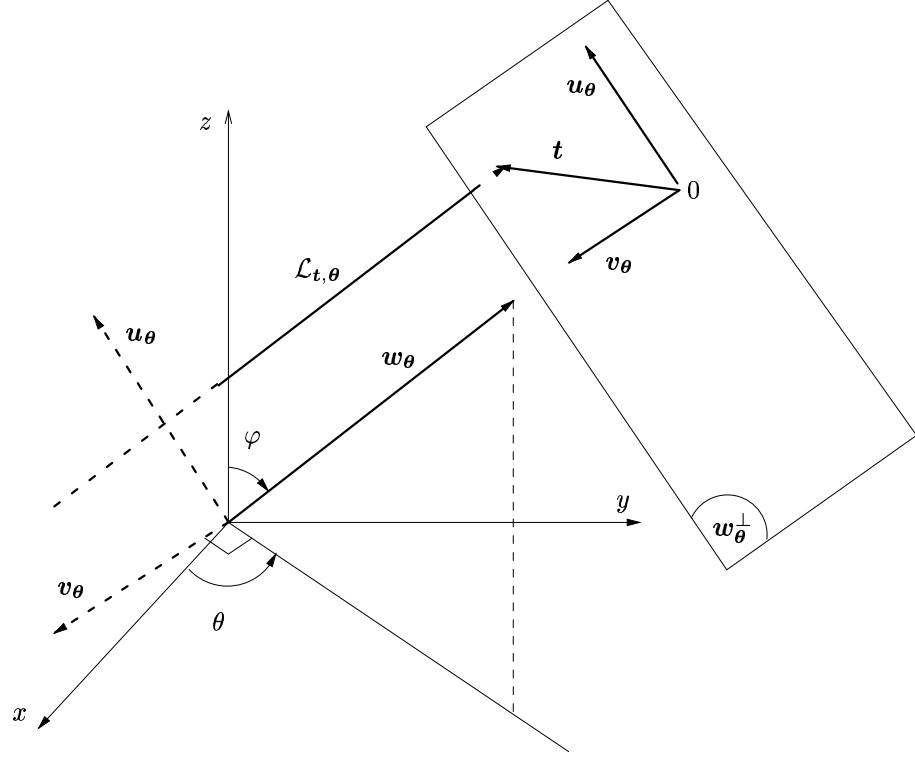


Figure I.4 – Mode de projection orthogonale en 3D.

Nous exprimons à présent la projection suivant la direction $\mathbf{w}_\theta$. Pour tout u et $v \in \mathbb{R}$, nous définissons $\mathbf{t} = u \mathbf{u}_\theta + v \mathbf{v}_\theta$ (point du plan $\mathbf{w}_\theta^\perp$) et nous calculons la projection suivant la droite $\mathcal{L}_{\mathbf{t},\theta}$ passant par ce point et dirigée par $\mathbf{w}_\theta$:

$$p_\theta(\mathbf{t}) = \iiint_{\mathbb{R}^3} f(x, y, z) \delta_2[\mathbf{t}_\theta(x, y, z) - \mathbf{t}] \, dx \, dy \, dz, \quad (\text{I.14})$$

où δ_2 est la distribution de Dirac 2D, et la fonction $\mathbf{t}_\theta : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ est définie par :

$$\begin{aligned} \forall (x, y, z) \in \mathbb{R}^3, \quad \mathbf{t}_\theta(x, y, z) &= [\langle \mathbf{OM}, \mathbf{u}_\theta \rangle, \langle \mathbf{OM}, \mathbf{v}_\theta \rangle]^t \\ &= \begin{pmatrix} -x \cos \varphi \cos \theta - y \cos \varphi \sin \theta + z \sin \varphi \\ x \sin \theta - y \cos \theta \end{pmatrix}. \end{aligned} \quad (\text{I.15})$$

Enfin, la k -ème projection de la scène ($1 \leq k \leq m$) est une image 2D constituée de $r_x \times r_y$ pixels. Elle correspond à l'échantillonnage de l'image continue $p_{\theta_k}(\mathbf{t})$:

$$\forall i \in \{0, \dots, r_x - 1\}, \forall j \in \{0, \dots, r_y - 1\}, \quad p_{i+1,j+1}^k = p_{\theta_k}(\mathbf{t}_k + (i \delta u_k, j \delta v_k)^t). \quad (\text{I.16})$$

Dans la suite, nous stockons cette k -ème projection sous la forme d'un vecteur de taille $r_x r_y$ obtenu en concaténant toutes les valeurs $p_{i+1,j+1}^k$.

I.2.3 Problème direct

En concaténant l'ensemble des données (toutes les projections) dans un vecteur $\mathbf{d}$ et en notant $\mathcal{A}$ l'opérateur de passage de la scène aux données, nous pouvons modéliser le problème direct par :

$$\mathbf{d} = \mathcal{A}_{\mathbf{t},\boldsymbol{\theta}}(\mathcal{S}) + \mathbf{b}, \quad (\text{I.17})$$

où la quantité $\mathbf{b}$, communément appelée « bruit », représente tous les facteurs que l'on ne maîtrise pas : les erreurs de mesure, les incertitudes du modèle et éventuellement les erreurs de discrétisation.

— Les erreurs de modèle sont dues à plusieurs facteurs : le caractère non ponctuel des sources et des détecteurs, le flux diffusé, le caractère non homogène de la scène, la non linéarité de la réponse du détecteur aux intensités reçues. En particulier, le détecteur est défini sous forme d'un ensemble de pixels et chacun d'eux reçoit un rayon X. Nous considérons qu'une mesure en un pixel particulier correspond à une *seule* direction de projection qui atteint le centre de ce pixel. Nous devrions en fait considérer l'ensemble des rayons qui atteignent la surface du pixel, et pondérer les intensités I_D correspondantes. Cette dernière approximation est acceptable dans le contexte du CND, car le nombre de pixels détecteurs est souvent très important (de l'ordre de 1024^2). Cela signifie alors que la surface des pixels est très faible.

— Les erreurs de discrétisation concernent le terme $\mathcal{A}(\mathcal{S})$. Elles dépendent de la modélisation choisie pour la scène. Par exemple, une modélisation par un ensemble de voxels génère des fonctions densité constantes par morceaux, les morceaux étant les domaines occupés par chaque voxel.

Comme précisé précédemment, nous ne maîtrisons pas le terme de bruit $\mathbf{b}$. C'est pourquoi nous le définissons comme un vecteur aléatoire que nous caractérisons par ses propriétés statistiques. L'hypothèse la moins informative et surtout la plus simple est de considérer $\mathbf{b}$ comme un vecteur aléatoire i.i.d., de loi gaussienne centrée de variance σ^2 .

I.3 Méthodes classiques de reconstruction

Nous avons vu que la reconstruction d'image en tomographie nécessite la modélisation de la scène par une carte de densité $f(x, y, z)$. Nous pouvons alors distinguer deux classes d'approches :

- Les méthodes analytiques qui sont basées sur l'équation intégrée de la transformée de Radon, et recherchent une expression de $f(x, y, z)$ à partir des projections $p_{\boldsymbol{\theta}_k}(\mathbf{t})$.
- Les méthodes algébriques qui discrétisent d'emblée la scène en voxels, et considèrent ensuite l'équation linéaire qui en découle.

Nous décrivons ces méthodes dans les paragraphes qui suivent, en insistant sur leurs limitations dans notre contexte. Cet exposé est réalisé dans le cadre bidimensionnel de façon à privilégier sa clarté, mais il peut être étendu au cas 3D (Feldkamp et al., 1984; Grangeat, 1987).

I.3.1 Méthodes analytiques

La recherche de la fonction $f(x, y)$ peut s'effectuer soit en passant dans le domaine de Fourier, soit directement en inversant la transformée de Radon. Nous décrivons brièvement ces deux méthodes dans les paragraphes ci-dessous.

I.3.1.1 Synthèse de Fourier

Il existe une relation fondamentale liant la transformée de Radon $p_\theta(t) = [\mathcal{A}f](t, \theta)$ et la fonction f elle-même. En notant indifféremment $\mathcal{F}$ les transformées de Fourier 1D et 2D, nous avons :

$$\forall \theta \in (0, 2\pi), \forall \nu \in \mathbb{R}_+, \mathcal{F}p_\theta(\nu) = \mathcal{F}f(\nu \cos \theta, \nu \sin \theta), \quad (\text{I.18})$$

où la transformée de Fourier qui intervient dans le terme de gauche est monodimensionnelle (Natterer, 1986). Dans cette formulation, nous ne considérons que les valeurs positives de la fréquence ν , à cause du caractère réel de la fonction p_θ ($\mathcal{F}p_\theta(-\nu) = \overline{\mathcal{F}p_\theta(\nu)}$). En supposant que le support de f soit compact ($\text{Supp}(f) \subset (-F/2, F/2)^2$), f est développable en série de Fourier sur son support. En tronquant ce développement sur N^2 termes, nous avons :

$$\forall (x, y) \in \text{Supp}(f), f(x, y) \approx \frac{1}{F^2} \sum_{j, k=-N/2}^{N/2-1} \mathcal{F}f(j/F, k/F) \exp\left(2i\pi \frac{jx + ky}{F}\right). \quad (\text{I.19})$$

Une méthode naturelle de reconstruction consiste donc à :

- Fixer N fréquences ν_i positives ;
- Calculer, pour tout k , la transformée de Fourier $\mathcal{F}d_k(\nu_i) \approx \mathcal{F}p_{\theta_k}(\nu_i)$ de la k -ème donnée en les fréquences ν_i ;
- Dédire de (I.19) les estimées des valeurs de $f(x, y)$ en les n pixels f_i .

La troisième étape pose un problème d'interpolation 2D. En effet, il s'agit d'estimer les quantités $\mathcal{F}f(j/F, k/F)$ à partir des mN termes $\mathcal{F}f(\nu_i, \theta_k)$. Les positions (ν_i, θ_k) sont sur des cercles concentriques (de rayons ν_i) et centrés en 0 dans le domaine de Fourier (voir la figure I.5). Il est clair que ces positions ne sont pas réparties uniformément dans le plan de Fourier, et que les erreurs d'interpolation sont d'autant plus importantes que $j^2 + k^2$ est important, c'est-à-dire que l'on s'éloigne de l'origine du plan de Fourier. Par conséquent, on limite généralement la valeur de N .

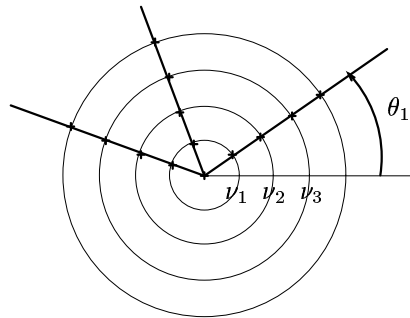


Figure I.5 – Représentation des données dans le plan de Fourier, *i.e.*, des signaux $\mathcal{F}d_k(\nu_i) \approx \mathcal{F}p_{\theta_k}(\nu_i)$. Le k -ème signal correspond à la demi-droite inclinée de θ_k par rapport à l'axe horizontal. La fonction $\mathcal{F}p_\theta(\nu)$ est mieux connue pour des fréquences proches de 0, car la densité de points (ν_i, θ_k) est plus importante dans cette région.

I.3.1.2 Inversion de la transformée de Radon

Les développements ci-dessus sont à la base de l'inversion de la transformée de Radon, et d'un algorithme rapide qui n'utilise pas l'interpolation fréquentielle précédente. À partir de la relation (I.18), nous pouvons obtenir la formule suivante, basée sur l'utilisation de l'opérateur de rétroprojection $\mathcal{A}^\#$:

$$f = \frac{1}{4\pi} \mathcal{A}^\# (\mathcal{I} \star \mathbf{d}), \quad (\text{I.20})$$

où le filtre $\mathcal{I}$ est le noyau de convolution dont la transformée de Fourier est $\mathcal{FI}(\nu) = |\nu|$. L'opérateur de rétroprojection agit sur l'espace de Schwartz des projections, $\mathcal{SC}(\mathbb{R}_+ \times (0, 2\pi))$. Il est défini par :

$$\begin{aligned} \mathcal{A}^\# : \quad \mathcal{SC}(\mathbb{R}_+ \times (0, 2\pi)) &\longrightarrow \mathcal{SC}(\mathbb{R}^2) \\ p &\longrightarrow [\mathcal{A}^\#(p)](M) = \int_0^{2\pi} p_\theta(\mathbf{w}_\theta^t M) d\theta, \quad \forall M \in \mathbb{R}^2. \end{aligned} \quad (\text{I.21})$$

La méthode d'inversion associée à (I.20) est relativement simple à mettre en œuvre. Elle consiste, pour chaque donnée $\mathbf{d}_k$, à :

- Calculer la transformée de Fourier $\mathcal{F}\mathbf{d}_k(\nu)$;
- Multiplier $\mathcal{F}\mathbf{d}_k(\nu)$ par $|\nu|/4\pi \rightarrow P_k(\nu)$, et calculer la transformée de Fourier inverse de $P_k(\nu)$ pour obtenir les projections filtrées $\mathcal{I} \star \mathbf{d}_k$;
- Faire la rétroprojection de $\mathcal{I} \star \mathbf{d}_k$;
- Mettre à jour l'image estimée $\hat{f}$.

Cet algorithme est couramment rencontré sous le nom de *rétroprojection filtrée*, car il consiste à effectuer successivement un filtrage des données et une rétroprojection. Il est très prisé en reconstruction d'images médicales car il permet de traiter les données $\mathbf{d}_k$ une par une, donc à chaque acquisition (cela ne nécessite pas la globalité des données pour commencer le traitement), et sa mise en œuvre est rapide, car à base d'algorithmes *FFT*.

I.3.1.3 Conclusions

Les méthodes précédentes ne fournissent pas de solutions acceptables dans notre contexte. En effet, elles souffrent directement du manque de données. En particulier, le caractère limité des angles de projection signifie que les angles $\theta_1, \dots, \theta_m$ sont tous dans un intervalle du type $(0, \Theta)$, où Θ est en général inférieur à $\pi/2$. Par conséquent, seule une portion très limitée du plan de Fourier est remplie par les données (voir la représentation fréquentielle de la figure I.5).

Indépendamment de ces arguments, nous mettons l'accent sur le fait que les méthodes analytiques requièrent nécessairement une discrétisation de la scène lors de l'évaluation de la solution $f(x, y)$. Pour cette raison, nous optons pour des méthodes dites algébriques, qui effectuent d'emblée cette discrétisation.

I.3.2 Méthodes algébriques

Principe. Ces techniques font appel à l'approximation de la scène par une fonction constante par morceaux. Dans le cas 2D, la scène est discrétisée en un ensemble de pixels, où le pixel est l'élément (rectangulaire) de surface élémentaire (Herman, 1980; Kak et Slaney, 1987). Si l'on divise la scène en une grille de n pixels de valeur $f_1, \dots, f_n$, l'expression (I.10) d'une projection devient :

$$p_\theta(t) = \mathbf{l}_\theta(t)^t \mathbf{f} = \sum_{j=1}^n a_j(t, \theta) f_j, \quad (\text{I.22})$$

où $\mathbf{f} = [f_1, \dots, f_n]^t \in \mathbb{R}_+^n$, et chaque terme $a_j(t, \theta)$ représente la longueur de l'intersection du j -ème pixel et de la droite $\mathcal{L}_{t, \theta}$ (voir la figure I.6). L'ensemble des projections $p_{\theta_k}(t_k + (i-1)\delta t_k)$ se réécrit matriciellement :

$$\mathbf{p}_\theta(t) = \mathbf{A} \mathbf{f}, \quad (\text{I.23})$$

où $\mathbf{p}_\theta(t)$ est le vecteur qui représente la projection de la scène et $\mathbf{A}$ est la matrice de projection, de taille $M \times n$. $\mathbf{A}$ est définie sous la forme de m blocs $\mathbf{A}_1, \dots, \mathbf{A}_m$ de matrices de taille $r \times n$:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \dots \\ \mathbf{A}_m \end{bmatrix}. \quad (\text{I.24})$$

Le bloc $\mathbf{A}_k$ correspond à la k -ème projection, son élément de ligne i et de colonne j est le terme $a_j(t_k + (i-1)\delta t_k, \theta_k)$. $\mathbf{A}_k$ dépend donc uniquement :

- de la pixellisation de la scène, et
- de la géométrie de la k -ème projection.

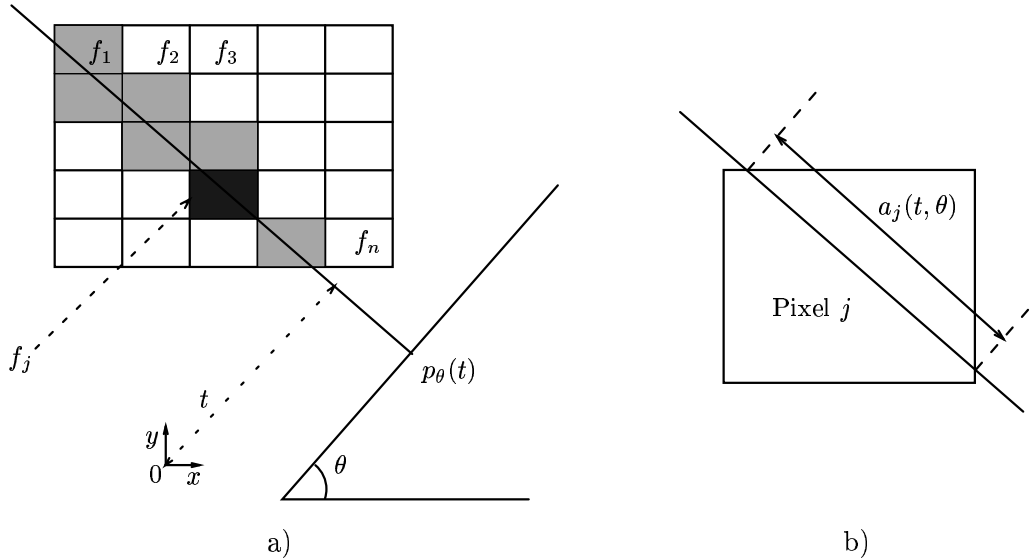


Figure I.6 – Projection d'une scène 2D pixellisée. a) Une mesure en (t, θ) . b) Contribution du j -ème pixel à cette mesure.

Inversion généralisée et décomposition en valeurs singulières. L'approche naturelle pour reconstruire la scène est de résoudre le système linéaire $\mathbf{d} = \mathbf{A}\mathbf{f}$. Comme $\mathbf{A}$ est en général une matrice rectangulaire, nous recherchons la solution au sens des moindres carrés :

$$\mathbf{f}_{\text{MC}} = \arg \min_{\mathbf{f}} \|\mathbf{d} - \mathbf{A}\mathbf{f}\|^2, \quad (\text{I.25})$$

ce qui équivaut à résoudre l'équation normale :

$$\mathbf{A}^t \mathbf{A} \mathbf{f} = \mathbf{A}^t \mathbf{d}, \quad (\text{I.26})$$

où la matrice $\mathbf{A}^t \mathbf{A}$ est de taille $n \times n$. Comme le nombre de projections est limité, nous avons $M < n$, et $\mathbf{A}^t \mathbf{A}$ n'est pas inversible (son rang est majoré par M). Par conséquent, le système (I.26) possède une infinité de solutions. Nous définissons la solution *inverse généralisée*, notée $\mathbf{f}_{\text{IG}}$, par la solution de (I.26) de norme minimale. En particulier, si $\mathbf{d} \in \text{Im}(\mathbf{A})$, l'unicité de $\mathbf{f}_{\text{IG}}$ est garantie.

En pratique, $\mathbf{f}_{\text{IG}}$ s'obtient en effectuant une décomposition en valeurs singulières de $\mathbf{A}$:

$$\mathbf{A} = \mathbf{U} \text{diag}[\lambda_1, \dots, \lambda_n] \mathbf{V}^t, \quad (\text{I.27})$$

où $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_n]$ et $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n]$ sont des matrices de taille $M \times n$ et $n \times n$, telles que $\mathbf{U}_M = \{\mathbf{u}_i, i = 1, \dots, M\}$ et $\mathbf{V}_n = \{\mathbf{v}_i, i = 1, \dots, n\}$ forment des bases orthonormées de $\mathbb{R}^M$ et $\mathbb{R}^n$, et les scalaires λ_i , appelés *valeurs singulières de $\mathbf{A}$* , forment une suite décroissante positive². La solution inverse généralisée s'obtient finalement à partir des bases $\mathbf{U}_M$ et $\mathbf{V}_n$ par :

$$\mathbf{f}_{\text{IG}} = \sum_{i=1}^{\text{rg}(\mathbf{A})} \frac{\mathbf{u}_i^t \mathbf{d}}{\lambda_i} \mathbf{v}_i. \quad (\text{I.28})$$

La stabilité de cette solution par rapport aux données est contrôlée par le nombre de condition de $\mathbf{A}$, *i.e.*, par le rapport $\lambda_1/\lambda_{\text{rg}(\mathbf{A})}$, et également par l'inverse du rapport signal à bruit des données. Dans notre situation, ces deux facteurs, et notamment le premier, sont très élevés. En effet, le fait que les projections soient en nombre limité et à angles limités induit un très mauvais conditionnement de $\mathbf{A}$. Cette remarque se rapproche de celles qui concernent le remplissage du domaine de Fourier pour les méthodes analytiques, et rend l'inverse généralisée très peu adaptée à notre situation.

I.3.3 Conclusions

Nous avons décrit dans cette partie les deux grandes classes de méthodes de reconstruction à partir de données tomographiques. Les algorithmes classiques nécessitent la connaissance « complète » (pour $\theta \in (0, 2\pi)$) des projections de $\mathbf{f}$, et produisent des résultats médiocres si l'information des données est limitée. Dans ce qui suit, nous proposons une structure dite *régularisée* qui permet de prendre en compte des hypothèses supplémentaires sur la scène de façon à rendre bien posé sa reconstruction.

Par ailleurs, nous privilégions les méthodes de type algébrique, où la scène est paramétrée d'emblée. Nous effectuerons au § I.5 de plus amples développements sur cette modélisation.

²Les vecteurs $\mathbf{v}_i$ sont des vecteurs propres de $\mathbf{A}^t \mathbf{A}$ associés aux valeurs propres λ_i^2 (en particulier $\lambda_i = 0$ pour $i > M$ car $\text{rg}(\mathbf{A}) \leq M$), et les vecteurs $\mathbf{u}_i$ sont tels que $\mathbf{A} \mathbf{v}_i = \lambda_i \mathbf{u}_i$. Par ailleurs, les vecteurs de $\mathbf{U}_M$ sont vecteurs propres de $\mathbf{A} \mathbf{A}^t$ associés aux valeurs propres $\lambda_1^2, \dots, \lambda_M^2$.

I.4 Méthodologie bayésienne

Nous avons vu dans la section précédente que les méthodes classiques de reconstruction ne mènent pas à des solutions convenables pour la présente étude. Sur la base de la formulation adoptée au § I.3.2, nous pouvons clairement identifier l'estimation de la scène à un problème inverse mal posé au sens de Hadamard (Tikhonov et Arsenin, 1976). En d'autres termes, l'une au moins des conditions suivantes n'est pas satisfaite :

1. Existence d'une solution $\hat{\mathcal{S}}(\mathbf{d})$ au problème inverse $\mathbf{d} = \mathcal{A}(\mathcal{S})$.
2. Unicité de cette solution.
3. Stabilité de cette solution, *i.e.*, continuité de la fonction $\mathbf{d} \rightarrow \hat{\mathcal{S}}(\mathbf{d})$.

En particulier, la solution inverse généralisée est définie de façon unique, mais n'est pas continue par rapport aux données. Une possibilité intéressante consiste alors à *régulariser*, en intégrant des informations *a priori* sur l'objet et/ou le fond homogène pour obtenir une solution convenable. Nous choisissons de modéliser les informations disponibles dans la formulation bayésienne. C'est une approche intéressante car elle permet de prendre en compte simultanément les informations *a priori* sur les paramètres inconnus $\mathcal{S}$ et sur les données $\mathbf{d}$, à travers des quantités probabilistes (resp. la loi *a priori* et la fonction de vraisemblance), puis de définir une distribution *a posteriori* sur $\mathcal{S}$. Elle consiste en les quatre points suivants :

- (a) Attribuer une distribution *a priori* $p(\mathcal{S})$ aux paramètres à estimer. Cela caractérise notre connaissance initiale de la scène.
- (b) Définir une distribution $p(\mathbf{d} | \mathcal{S})$ propre aux grandeurs mesurées. Cette loi, vue comme une fonction de $\mathcal{S}$, est appelée fonction de *vraisemblance* de la scène. Elle traduit notre imprécision sur les données.
- (c) Fusionner les deux informations précédentes à l'aide de la règle de Bayes pour définir la distribution *a posteriori* $p(\mathcal{S} | \mathbf{d})$.
- (d) Choisir un estimateur $\hat{\mathcal{S}}$ des paramètres à partir de $p(\mathcal{S} | \mathbf{d})$.

Pour notre application, nous n'avons pas d'information générale très précise quant à la forme de $\mathcal{S}$. La distribution $p(\mathcal{S})$ portera sur l'homogénéité de la scène, et dépendra de la modélisation choisie. Nous laissons le point (a) en suspens dans ce chapitre. Compte-tenu du modèle direct (I.17), la fonction de vraisemblance est donnée par la loi du bruit $\mathbf{b}$, que nous avons supposée gaussienne. Nous avons donc :

$$p(\mathbf{d} | \mathcal{S}) = \mathcal{N}(\mathbf{p}(\mathcal{S}), \sigma^2 \mathbf{I}_M), \quad (\text{I.29})$$

où M est la taille des données et $\mathbf{I}_M$ est la matrice unitaire de taille $M \times M$. La règle de Bayes s'énonce ainsi :

$$p(\mathcal{S} | \mathbf{d}) = \frac{p(\mathbf{d} | \mathcal{S}) p(\mathcal{S})}{p(\mathbf{d})}, \quad (\text{I.30})$$

où $p(\mathbf{d})$ est le facteur de normalisation (indépendant de l'inconnu $\mathcal{S}$) défini par :

$$p(\mathbf{d}) = \int_{\mathcal{S}} p(\mathbf{d} | \mathcal{S}) p(\mathcal{S}) d\mathcal{S}. \quad (\text{I.31})$$

Dans tout ce qui suit, nous estimons la scène (opération (d)) au sens du maximum *a posteriori* (MAP). En d'autres termes, nous maximisons la fonctionnelle $p(\mathcal{S} | \mathbf{d})$, ou minimisons l'anti « log-vraisemblance » *a posteriori* :

$$\begin{aligned}
\hat{\mathcal{S}} &= \arg \min_{\mathcal{S}} \left\{ -\log p(\mathcal{S} | \mathbf{d}) \right\} = \arg \min_{\mathcal{S}} \left\{ -\log p(\mathbf{d} | \mathcal{S}) p(\mathcal{S}) \right\} \\
&= \arg \min_{\mathcal{S}} \left\{ \mathcal{J}(\mathcal{S}) = \|\mathbf{d} - \mathcal{A}(\mathcal{S})\|^2 - (2\pi\sigma^2)^M \log p(\mathcal{S}) \right\}.
\end{aligned} \tag{I.32}$$

Nous minimisons ainsi l'énergie *a posteriori* $\mathcal{J}(\mathcal{S}) = \mathcal{K}(\mathcal{S}) + \lambda \mathcal{R}(\mathcal{S})$, fonctionnelle définie par la somme d'un terme d'attache aux données :

$$\mathcal{K}(\mathcal{S}) = \|\mathbf{d} - \mathcal{A}(\mathcal{S})\|^2, \tag{I.33}$$

et d'un terme dit de *régularisation*, défini par $\mathcal{R}(\mathcal{S}) \propto -\log p(\mathcal{S})$, et qui prend en compte les informations *a priori* sur la scène. Le compromis entre ces deux termes est géré par un paramètre de régularisation λ . Compte-tenu de nos choix ultérieurs pour $\mathcal{S}$ et sa distribution *a priori*, la fonction $\mathcal{J}$ est continue, positive et infinie quand les paramètres du modèle tendent vers l'infini. Elle est convexe ou non suivant les modèles. Si la formulation (I.32) garantit l'existence d'un minimum global de $\mathcal{J}$, son unicité n'est pas acquise dans les cas où $\mathcal{J}$ est non convexe.

I.5 Vers la modélisation de la scène

Nous précisons à présent les différents types de modèles appropriés à notre problème. À cet effet, nous rappelons que la scène est *binaire*, et constituée d'un objet *compact* dans un fond homogène. Nous distinguons deux grandes classes de modèles :

1. Ceux qui définissent l'image par un ensemble de voxels binaires, où le voxel est l'élément de volume élémentaire (de forme parallélépipédique en général).
2. Ceux qui modélisent directement le contour ou la surface de l'objet (surface fermée)³.

Approche par voxels. Cette approche est la plus classique, et a été largement utilisée en imagerie médicale (Herman, 1980), en CND (Klifa, 1991; Sauer et al., 1994; Gautier, 1996; Robini, 1998; Chalmond et al., 1999) et en détonique, pour la reconstruction d'objets en déformation à partir de radiographies (Dinten, 1990). Cette approche présente l'avantage de mener à un problème direct linéaire (voir le § I.3.2), ce qui conduit à des méthodes de reconstruction relativement simples.

Approche par contour. La seconde approche est plus adaptée à notre problème, car elle prend directement en compte le caractère compact de l'objet (frontière fermée), sans modéliser explicitement la totalité du volume de la scène. Elle permet donc une réduction très nette du nombre de paramètres du modèle (nous verrons que ce nombre se chiffre à quelques centaines contre quelques millions pour les méthodes par voxels). En revanche, l'opérateur de projection n'est pas linéaire par rapport aux paramètres, ce qui accroît la complexité de la reconstruction.

Cette classe de modèles a été largement utilisée en segmentation d'images et en vision par ordinateur, où l'on recherche à visualiser la forme d'un objet à partir de données projectives (*design* de contour ou de surface). En revanche, son utilisation pour la reconstruction tomographique a été plus restreinte, car les données ne sont pas reliées de façon simple aux paramètres du modèle. Néanmoins, les modèles par contours ont été utilisés avec succès dans certaines modalités d'imagerie médicale (Cinquin, 1981; Fessler et Macovski, 1991). Bien qu'en général, ces applications reposent sur un nombre très important de projections⁴, il existe des situations pour lesquelles le nombre de vues est faible, de façon à limiter la dose d'irradiation délivrée au patient. C'est notamment le cas en angiographie, où de nombreux travaux concernent la reconstruction d'images binaires à partir d'un nombre très

³En général, nous utiliserons les dénominations *contours* déformables en 2D et *surfaces* déformables en 3D.

⁴Les méthodes classiques (par voxels) de type rétroprojection filtrée fournissent alors de très bonnes reconstructions.

limité de vues (Fessler et Macovski, 1991; Pellot et al., 1994; Coutand et al., 1997; Senasli et al., 2000). Dans cette application, il apparaît primordial d'exploiter la très bonne connaissance *a priori* des zones à reconstruire (qui sont les organes du patient) pour fournir un modèle spécifique de type contour déformable à très faible nombre de paramètres.

Approche mixte. Enfin, il existe des approches «mixtes», qui modélisent des images formées de zones homogènes (à faibles variations de densités) mais non uniformes (Coutand et al., 1997; Battle et al., 1998). Pour cela, il est possible d'utiliser simultanément les deux approches précédentes. On définit un modèle par contour de façon à spécifier des régions d'intérêt bien identifiées, puis on discrétise ces zones en un ensemble de voxels. La reconstruction de la scène consiste généralement à des estimations alternatives des contours et des voxels qui constituent leur intérieur ou leur extérieur.

I.6 Conclusions

Dans ce chapitre introductif, nous avons posé le problème de la reconstruction d'objets compacts binaires à partir d'un faible nombre de projections (dans un contexte où les angles sont limités). Nous avons montré que ce problème est *mal posé*, et que la reconstruction nécessite de *régulariser*, en intégrant des informations *a priori* sur la scène. Leur nature dépend du modèle choisi pour la scène. Nous avons défini deux grandes classes de modèles, que nous développons dans les chapitres suivants. Au chapitre II, nous étudions le modèle par voxels et les techniques d'estimation associées. Le chapitre III est une introduction aux modèles par contours déformables, qui apparaissent particulièrement adaptés à notre problème. Parmi eux, nous sélectionnons un contour paramétrique spécifique (modèle par spline fermée d'ordre 1), que nous développons aux chapitres IV et V. La structure de la reconstruction fait intervenir l'estimation des positions des sommets du modèle. Nous montrons en particulier que ces contours offrent un bon compromis entre la variété des formes qu'ils génèrent et la rapidité de la reconstruction.

CHAPITRE II

APPROCHE PAR VOXELS

- II.1 Modélisation du problème
- II.2 Construction du critère régularisé
- II.3 Optimisation du critère
- II.4 Réduction du nombre de voxels
- II.5 Stratégie multirésolution
- II.6 Conclusions
- II.A Preuves des propositions 1 et 2

II.1 Modélisation du problème

Dans ce chapitre, nous nous plaçons exclusivement dans le cas 3D. L'objet n'est pas de développer une nouvelle méthode de reconstruction par voxels, mais de faire une synthèse des méthodes classiques pour proposer une méthode concurrente aux méthodes par contours à des fins de comparaison. Nous modélisons la scène $\mathcal{S}$ par un ensemble de n voxels binaires $\mathbf{f} = [f_1, \dots, f_n]^t$, où le voxel est l'élément de volume élémentaire. En général, on choisit par simplicité des voxels de forme identique et cubique. La binarité s'exprime de la façon suivante : $f_i = 1$ si et seulement si le centre du i -ème voxel est à l'intérieur de l'objet.

II.1.1 Remarques préliminaires

Avant d'évoquer l'estimation de $\mathbf{f}$, nous revenons brièvement sur la structure de l'opérateur de projection. Cet opérateur est linéaire et s'écrit :

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \dots \\ \mathbf{p}_m \end{bmatrix} = \mathbf{A} \mathbf{f} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \dots \\ \mathbf{A}_m \end{bmatrix} \mathbf{f}, \quad (\text{II.1})$$

où $\mathbf{A}$ est la matrice de projection, de taille $M \times n$. Plus précisément, le vecteur $\mathbf{p}$ et la matrice $\mathbf{A}$ sont définis par blocs de projections. Prenons la k -ème projection : le vecteur $\mathbf{p}_k$ est de taille $r_x r_y$ et la matrice $\mathbf{A}_k$ est de taille $(r_x r_y) \times n$. $\mathbf{A}$ est donc de taille énorme (d'autant plus grande que la résolution des détecteurs et/ou du modèle est fine), et possède en général un plus grand nombre de colonnes que de lignes, d'où le caractère mal conditionné du problème. Nous soulignons toutefois que $\mathbf{A}$ est une *matrice creuse* dans notre contexte. En effet, ses éléments sont les longueurs des intersections des rayons de projection et des voxels de $\mathbf{f}$ (en 2D, ce sont les termes $a_j(t, \theta)$ de la figure I.6 (b)), et chaque rayon de projection n'intersecte qu'un faible nombre de voxels.

II.1.2 Estimation de la scène

Compte tenu du caractère mal conditionné de $\mathbf{A}$ et des développements du chapitre I (§ I.4), nous régularisons le problème et ramenons la reconstruction de la scène « voxellisée » à la minimisation d'une énergie *a posteriori* :

$$\begin{aligned}\mathcal{J}(\mathbf{f}) &= \mathcal{K}(\mathbf{f}) + \lambda \mathcal{R}(\mathbf{f}) \\ &= \|\mathbf{d} - \mathbf{A}\mathbf{f}\|^2 + \lambda \mathcal{R}(\mathbf{f}),\end{aligned}\tag{II.2}$$

où les opérateurs $\mathcal{K}$ et $\mathcal{R}$ tiennent compte respectivement de l'adéquation du modèle aux données et de l'information *a priori* sur $\mathbf{f}$. Il est simple de choisir un potentiel $\mathcal{R}$ qui favorise la formation de régions homogènes (cf. § II.2), mais les informations « un seul objet » et « objet compact » (frontière fermée) sont complexes à modéliser explicitement. L'estimée au sens du MAP est notée $\hat{\mathbf{f}}$, et minimise le critère $\mathcal{J}(\mathbf{f})$:

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f} \in \{0,1\}^n} \mathcal{J}(\mathbf{f}).\tag{II.3}$$

Il existe plusieurs façons d'envisager cette minimisation :

- Tenir compte explicitement de l'hypothèse $\hat{\mathbf{f}} \in \{0,1\}^n$ et minimiser directement $\mathcal{J}$ sur ce domaine. Cela relève de la *tomographie discrète*, et mène à des problèmes complexes de combinatoire. Le lecteur est renvoyé à (Kuba et Herman, 1999) pour une synthèse bibliographique.
- Étendre, puis minimiser le critère $\mathcal{J}$ sur des domaines continus : $\mathbb{R}^n$, $\mathbb{R}_+^n$ ou l'intervalle $(0,1)^n$.

Par simplicité, nous choisissons la seconde option, et minimisons $\mathcal{J}$ sur l'intervalle $(0,1)^n$.

II.1.3 Organisation du chapitre

Dans la suite, nous développons la méthode introduite ci-dessus pour l'estimation directe de $\mathbf{f}$ à partir des données, basée sur la minimisation de $\mathcal{J}(\mathbf{f})$ sur $(0,1)^n$. La section II.2 est consacrée au choix de l'énergie *a priori* $\mathcal{R}(\mathbf{f})$, et la section II.3 développe l'algorithme de minimisation associé (algorithme du gradient projeté). Enfin, nous verrons aux § II.4 et II.5 comment limiter le coût algorithmique en proposant une stratégie multirésolution.

II.2 Construction du critère régularisé

Cette section est consacrée à la construction du critère $\mathcal{J} = \mathcal{K} + \lambda \mathcal{R}$, et plus particulièrement de la fonctionnelle $\mathcal{R}$. Tout d'abord, nous notons que $\mathcal{K}$ est une énergie *quadratique* en $\mathbf{f}$:

$$\mathcal{K}(\mathbf{f}) = \|\mathbf{d} - \mathbf{A}\mathbf{f}\|^2.\tag{II.4}$$

Pour favoriser la formation de régions homogènes, le choix naturel pour $\mathcal{R}(\mathbf{f})$ est un potentiel markovien, qui pénalise les différences entre voxels adjacents. La loi *a priori* est de la forme :

$$p(\mathbf{f}) \propto \exp\left[-\frac{1}{T} \sum_{i \sim j} \phi(f_i - f_j)\right],\tag{II.5}$$

où :

- « $\sim$ » dénote la relation de 6-voisinage entre voxels ;
- $\phi : \mathbb{R} \rightarrow \mathbb{R}_+$ est une fonction continue, paire et croissante sur $\mathbb{R}_+$.

D'après les développements du chapitre I, nous pouvons écrire :

$$\mathcal{R}(\mathbf{f}) = \mathcal{D}(\mathbf{f}) = \sum_{i \sim j} \phi(f_i - f_j),\tag{II.6}$$

en définissant le paramètre de régularisation par $\lambda = (2\pi\sigma^2)^M/T \geq 0$. Le problème auquel nous sommes confrontés est *le choix de la fonction ϕ* . Compte-tenu du caractère quadratique donc convexe de $\mathcal{K}$, une fonction ϕ convexe rend la fonction $\mathcal{R}$, et donc finalement le critère $\mathcal{J} = \mathcal{K} + \lambda\mathcal{R}$ convexe par rapport à $\mathbf{f}$. Ceci présente l'avantage de faciliter l'optimisation de $\mathcal{J}$, mais au détriment de la qualité de la reconstruction. À présent, nous présentons une synthèse bibliographique à propos du choix du potentiel ϕ , et des techniques associées pour la minimisation de $\mathcal{J}$.

II.2.1 Choix d'un potentiel convexe

Le choix d'une fonction ϕ convexe mène à des algorithmes d'optimisation simples pour $\mathcal{J}$, en général de type gradient. En revanche, la convexité de ϕ impose que $\phi(t) \rightarrow \infty$ quand $t \rightarrow \infty$, ce qui limite la qualité des reconstructions en pénalisant trop les discontinuités. Le choix le plus simple est un potentiel quadratique ($\phi(t) = t^2$, *i.e.*, régularisation $\mathcal{L}_2$). Il correspond à une distribution *a priori* $p(\mathbf{f})$ gaussienne (compte tenu des hypothèses sur le bruit), et confère à la fonctionnelle $\mathcal{J}$ un caractère quadratique. Par conséquent, son minimiseur global s'exprime linéairement en fonction des données. En pratique, compte tenu de la grande taille de la matrice $\mathbf{A}$, des algorithmes itératifs sont préconisés. En dépit de cette simplicité, la pénalisation $\mathcal{L}_2$ est peu adaptée à la reconstruction d'images homogènes par morceaux car elle pénalise trop l'apparition de bords francs. Elle mène donc à des images lissées.

Un autre choix possible est la régularisation $\mathcal{L}_1$ ($\phi(t) = |t|$), qui correspond à un *a priori* laplacien. Ce choix pénalise beaucoup moins les fortes valeurs des différences entre voxels, mais le caractère non différentiable de ϕ , et donc de $\mathcal{J}$ en certains points, pose des problèmes quant à l'utilisation d'algorithmes classiques d'optimisation (de type gradient).

Il existe des pénalisations «intermédiaires» qui préservent les bords tout en favorisant les zones homogènes. Bouman et Sauer (1993) ont étudié les modèles *gaussiens généralisés* (GMRF, *i.e.*, régularisation $\mathcal{L}_\gamma$), correspondant à $\phi_\gamma(t) = |t|^\gamma$, $1 < \gamma \leq 2$ (voir figure II.1 (a)). Citons également les fonctions $\mathcal{L}_2$ - $\mathcal{L}_1$ (quadratiques en 0, linéaires en l'infini), par exemple la fonction $\phi_T(t) = \sqrt{t^2 + T^2} - T$, $T > 0$ utilisée par (Gautier, 1996), ou la fonction de Huber (Huber, 1981) définie par :

$$\phi_T(t) = \begin{cases} t^2/T^2 & \text{si } |t| < T, \\ 2|t|/T - 1 & \text{sinon.} \end{cases} \quad (\text{II.7})$$

Cette dernière fonction est représentée en figure II.1 (b). Elle est définie par morceaux de façon simple, en raccordant une fonction quadratique pour $|t| < T$ et une fonction linéaire pour $|t| > T$. Elle favorise donc la reconstruction de bords francs.

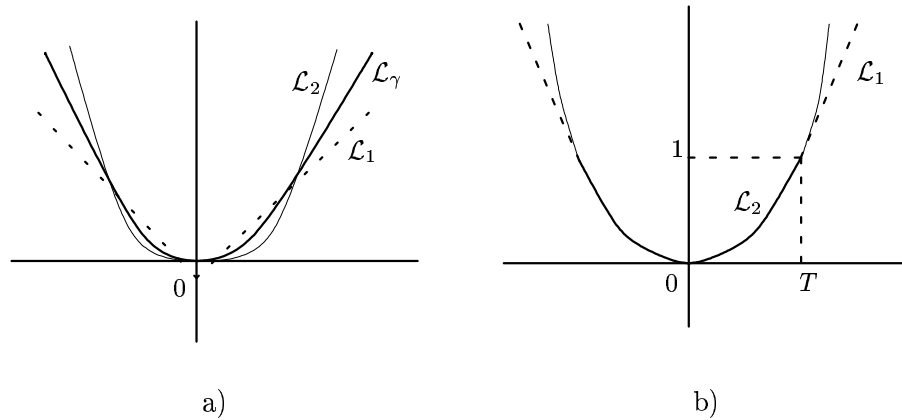


Figure II.1 – Choix classiques de fonction ϕ convexe pour le premier terme de régularisation.

a) Régularisations par des lois gaussienne, laplacienne ou gaussienne généralisée. b) Potentiel de Huber (en gras), *i.e.*, fonction $\mathcal{L}_2$ - $\mathcal{L}_1$.

II.2.2 Choix d'un potentiel non convexe

Un potentiel non convexe autorise plus facilement les discontinuités dans l'image, en bornant la fonction $\phi(t)$. L'exemple le plus simple est la quadrique tronquée, illustrée en figure II.2 (a) :

$$\phi_T(t) = \min(1, t^2/T^2). \quad (\text{II.8})$$

Cette fonction n'est pas dérivable en $t = -T$ et $t = T$, on lui préfère en général des fonctions de classe $\mathcal{C}^1$, par exemple $\phi(t) = -1/(1 + t^2/T^2)$ ou $\phi(t) = \ln(1 + t^2/T^2)$. Enfin, les régularisations entropiques (Mohammad-Djafari et Demoment, 1988), qui mènent à l'estimateur dit à maximum d'entropie :

$$\phi(t) = -|t| \ln |t|, \quad (\text{II.9})$$

sont particulièrement adaptées à la reconstruction d'images uniformes par morceaux et à bords francs.

Le caractère non convexe du potentiel ϕ mène à un critère $\mathcal{J}(\mathbf{f})$ non convexe et en général multimodal. Pour envisager son optimisation, il existe plusieurs options :

1. Utiliser un algorithme stochastique d'optimisation de fonctionnelle non convexe, de type *recuit simulé* (voir (Geman et Geman, 1984) en segmentation d'images, (Robini, 1998) en reconstruction tomographique). C'est un algorithme itératif qui échantillonne successivement les lois de probabilité

$$p_{\tau_k}(\mathbf{f} | \mathbf{d}) \propto \exp(-\mathcal{J}(\mathbf{f})/\tau_k),$$

où τ_k est une suite de *températures* qui décroît vers 0 de façon suffisamment lente. À la k -ème itération, on parcourt l'ensemble des voxels N_k fois et on échantillonne la valeur $f_i^{(k)}$ de chaque voxel suivant la loi :

$$p_{\tau_k}(f_i = \xi | f_j^{(k-1)}, j \neq i, \mathbf{d}) = p_{\tau_k}(f_i = \xi | f_j^{(k-1)}, j \sim i, \mathbf{d}).$$

C'est un échantillonnage de Gibbs compte tenu de la régularisation $\mathcal{D}(\mathbf{f})$ choisie. Il existe des résultats de convergence vers le minimum global de $\mathcal{J}$, mais leurs hypothèses nécessitent une décroissance très lente de la suite (τ_k) , donc un temps de calcul très élevé.

2. Avoir recours à une stratégie déterministe sous-optimale de type ICM (*iterated conditional modes*, Besag, 1986), éventuellement couplée avec un algorithme multirésolution (Bouman et Liu, 1988). Cet algorithme, également connu sous le nom de Gauss-Seidel en analyse numérique, consiste à visiter N fois l'ensemble des voxels, et à faire, à la k -ème visite du voxel i :

$$f_i^{(k)} = \arg \max_{f_i} \left\{ p(f_i | f_j^{(k-1)}, j \neq i, \mathbf{d}) = p(f_i | f_j^{(k-1)}, j \sim i, \mathbf{d}) \right\},$$

c'est-à-dire que l'on assigne à chaque voxel le mode de la distribution *a posteriori* conditionnelle correspondante. Cet algorithme converge car la suite $\mathcal{J}(\mathbf{f}^{(k)})$ est décroissante et positive. Si la convergence est bien plus rapide qu'un algorithme de recuit, nous n'avons en revanche pas de garantie de convergence vers le minimum global de $\mathcal{J}$.

Le couplage avec un algorithme multirésolution se justifie par le fait qu'aux résolutions les plus grossières, le nombre de minima locaux de $\mathcal{J}$ est faible, et l'estimateur MAP peut fournir une initialisation aux résolutions supérieures qui se trouve dans le bassin d'attraction du minimum global de $\mathcal{J}$.

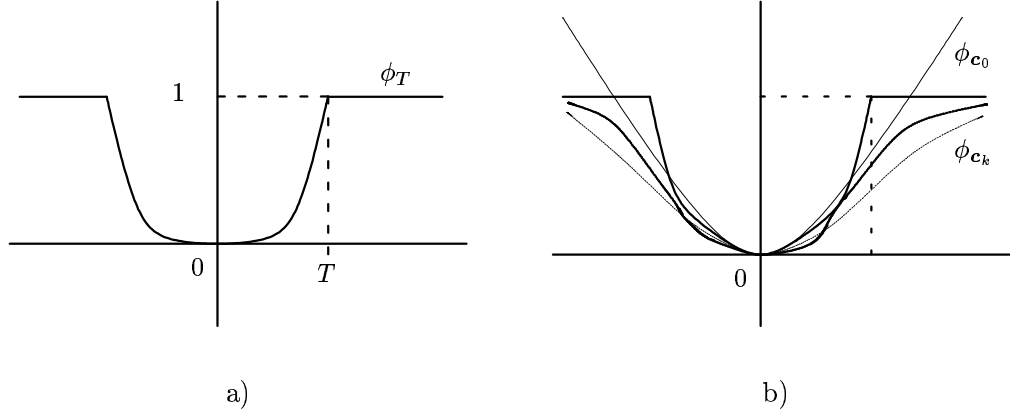


Figure II.2 – Principe du GNC. a) Régularisation par une loi gaussienne tronquée (fonction $\mathcal{L}_2\text{--}\mathcal{L}_0$). b) Fonctions ϕ_{c_k} approximant ϕ .

3. Utiliser les techniques de *Mean Field Annealing* (Bilbro et al., 1992) ou de non convexité graduelle (GNC). Cette dernière a été introduite par Blake et Zisserman (1987) en segmentation d'images, puis étendue aux problèmes inverses par Nikolova (1999). Son fondement est le *principe de continuation* ; pour effectuer la minimisation de $\mathcal{J}$, il s'agit de définir une suite de fonctionnelles $\mathcal{J}_{c_k}$ dépendant d'un petit nombre de paramètres de relaxation c_k par :

$$\mathcal{J}_{c_k}(\mathbf{f}) = \mathcal{K}(\mathbf{f}) + \lambda \sum_{i \sim j} \phi_{c_k}(f_i - f_j).$$

L'algorithme du GNC consiste à minimiser successivement les fonctionnelles $\mathcal{J}_{c_k}$ avec un algorithme local d'optimisation dont l'initialisation est la solution $\mathbf{f}^{(k-1)}$ à l'itération $k - 1$. Les fonctions ϕ_{c_k} sont choisies de telle sorte que ϕ_{c_0} est convexe et pour tout t , $\phi_{c_k}(t) \rightarrow \phi(t)$ quand $k \rightarrow \infty$. Sur la figure II.2 (b), nous illustrons la régularisation $\mathcal{L}_2\text{--}\mathcal{L}_0$, où la fonction ϕ est la quadrique tronquée définie en (II.8).

II.2.3 Choix du critère

Second terme de régularisation. Par simplicité, nous optons pour une fonction ϕ convexe et différentiable. Les conditions de convergence de l'algorithme utilisé (voir le § II.3 et l'annexe A) mènent à une fonction $\mathcal{L}_2\text{--}\mathcal{L}_1$, par exemple la fonction de Huber ϕ_T . De plus, nous exploitons l'hypothèse de binarité en rajoutant un second terme de régularisation $\mathcal{U}(\mathbf{f})$ qui rappelle les voxels à 0, c'est-à-dire au fond homogène, en tenant compte du fait que le défaut ne concerne qu'une minorité des pixels (Gautier, 1996). Le critère régularisé s'écrit finalement :

$$\begin{aligned} \mathcal{J}(\mathbf{f}) &= \mathcal{K}(\mathbf{f}) + \lambda \mathcal{D}(\mathbf{f}) + \mu \mathcal{U}(\mathbf{f}) \\ &= \mathcal{K}(\mathbf{f}) + \lambda \sum_{i \sim j} \phi_T(f_i - f_j) + \mu \sum_i \psi(f_i), \end{aligned} \quad (\text{II.10})$$

où ψ est un potentiel convexe et $\mathcal{C}^1$. Le critère $\mathcal{J}(\mathbf{f})$ reste convexe comme somme de trois fonctions convexes, et son optimisation sur $(0, 1)^n$ est menée par l'algorithme du *gradient projeté*, développé au § II.3.

Choix du potentiel ψ . Par simplicité, nous considérons les pénalisation $\mathcal{L}_\gamma$, $\gamma \geq 1$, i.e., $\psi_\gamma(t) = |t|^\gamma$. Les valeurs de γ proches de 1 sont intéressantes, car elles permettent de pénaliser faiblement les valeurs non nulles des voxels. Le choix $\gamma = 1$ est problématique, menant à un critère non différentiable. En revanche, le remplacement de la fonction valeur absolue (fonction ψ_1) par la fonction identité est envisageable, en tenant compte du fait que nous minimisons $\mathcal{J}$ sur $(0, 1)^n$ et non $\mathbb{R}^n$. Le critère (II.10) est alors différentiable sur $(0, 1)^n$ et s'écrit finalement :

$$\mathcal{J}(\mathbf{f}) = \mathcal{K}(\mathbf{f}) + \lambda \sum_{i \sim j} \phi_T(f_i - f_j) + \mu \sum_i f_i. \quad (\text{II.11})$$

Réglage des hyperparamètres. La formulation (II.11) impose le réglage de 3 hyperparamètres λ , μ et T . Le choix des valeurs de ces paramètres est empirique. En suivant les travaux de Gautier (1996), nous fixons dans un premier temps λ , puis optons pour une faible valeur de μ de l'ordre de $\lambda/6$. Le paramètre T joue le rôle de paramètre de seuil pour la détection de voxels adjacents homogènes. Compte-tenu de la gamme des valeurs des voxels, T peut être fixé indépendamment de λ et μ à une valeur de l'ordre de $1/5$.

II.3 Optimisation du critère

Nous minimisons le critère convexe $\mathcal{J}$ (fonction $\mathcal{C}^1$) défini en (II.11) sur le domaine $\mathcal{X} = (0, 1)^n$ par un algorithme itératif de descente de type *gradient projeté* sur l'ensemble des contraintes (Bertsekas, 1995, p. 203). Cet algorithme est une adaptation de l'algorithme du gradient dans le cas où l'on minimise sur un sous-domaine de $\mathbb{R}^n$ convexe et fermé. Il mène à un minimum local de $\mathcal{J}$ sur $\mathcal{X}$. Comme $\mathcal{J}$ est convexe, $\mathcal{J}$ possède sur $\mathcal{X}$ un seul minimum local qui est son minimum global $\mathbf{f}^*$.

II.3.1 Principe de l'algorithme

Tout d'abord, nous définissons l'opérateur de projection sur le domaine $\mathcal{X}$ par :

$$[\cdot]^+ : \mathbb{R}^n \longrightarrow \mathcal{X}$$

$$\forall \mathbf{f} \in \mathbb{R}^n, [\mathbf{f}]^+ = \arg \min_{\mathbf{g} \in \mathcal{X}} \|\mathbf{g} - \mathbf{f}\|^2 = \arg \min_{\mathbf{g} \in \mathcal{X}} \sum_{i=1}^n (g_i - f_i)^2. \quad (\text{II.12})$$

Compte tenu de la définition de $\mathcal{X}$, (II.12) est clairement équivalent à :

$$\forall \mathbf{f} \in \mathbb{R}^n, \forall i \in \{1, \dots, n\}, f_i^+ = \begin{cases} f_i & \text{si } f_i \in (0, 1), \\ 1 & \text{si } f_i > 1, \\ 0 & \text{sinon,} \end{cases} \quad (\text{II.13})$$

en notant f_i^+ les composantes de $[\mathbf{f}]^+$. La projection sur $\mathcal{X}$ revient donc à faire un seuillage des valeurs des voxels de l'image $\mathbf{f}$ sur $(0, 1)$.

L'algorithme de *gradient projeté* est défini itérativement par les relations :

$$\mathbf{f}^{(k+1)} = \mathbf{f}^{(k)} + \alpha_k (\bar{\mathbf{f}}^{(k)} - \mathbf{f}^{(k)}), \quad (\text{II.14})$$

$$\bar{\mathbf{f}}^{(k)} = [\mathbf{f}^{(k)} - s_k \nabla \mathcal{J}(\mathbf{f}^{(k)})]^+, \quad (\text{II.15})$$

où :

- $\alpha_k \in (0, 1)$ sont des paramètres de relaxation.
- s_k sont des paramètres positifs.

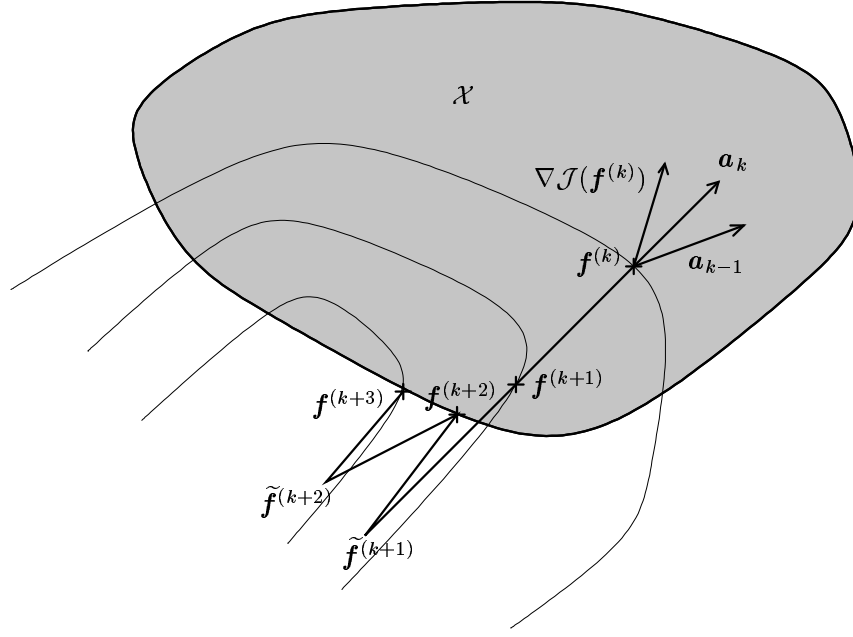


Figure II.3 – Illustration de la méthode du gradient projeté pour la minimisation de $\mathcal{J}(\mathbf{f})$ sur $\mathcal{X} \subset \mathbb{R}^n$. Les minimisations de ligne sont effectuées suivant les directions $\mathbf{a}_k$ des bissectrices de $\mathbf{a}_{k-1}$ et $\nabla \mathcal{J}(\mathbf{f}^{(k)})$.

L'appartenance de $\mathbf{f}^{(k)}$ et $\bar{\mathbf{f}}^{(k)}$ à $\mathcal{X}$ et la convexité de $\mathcal{X}$ garantissent l'appartenance de $\mathbf{f}^{(k+1)}$ à $\mathcal{X}$. Il y a deux options possibles pour le choix des paramètres de l'algorithme :

(i) Fixer $s_k = s$, $\forall k$ et choisir α_k de telle sorte que :

$$\alpha_k = \arg \min_{\alpha \in (0,1)} \mathcal{J}(\mathbf{f}^{(k)} + \alpha (\bar{\mathbf{f}}^{(k)} - \mathbf{f}^{(k)})). \quad (\text{II.16})$$

(ii) Fixer $\alpha_k = 1$, $\forall k$, et choisir une suite (s_k) constante ou non. Les équations (II.14) et (II.15) se simplifient en :

$$\mathbf{f}^{(k+1)} = \bar{\mathbf{f}}^{(k)} = [\mathbf{f}^{(k)} - s_k \nabla \mathcal{J}(\mathbf{f}^{(k)})]^+, \quad (\text{II.17})$$

c'est-à-dire que l'estimée à l'itération $k+1$ est la projection sur $\mathcal{X}$ de l'estimée que l'on obtiendrait par une méthode de descente non contrainte.

II.3.2 Convergence

Il existe des garanties de convergence de l'algorithme du gradient projeté (Bertsekas, 1995) dans les cas suivants :

- Cas (i), quelle que soit la valeur de $s > 0$.
- Cas (ii), pour le choix d'une suite (s_k) telle que :

$$s_k \rightarrow 0 \text{ quand } k \rightarrow \infty, \quad \text{et} \quad \sum_{k \geq 0} s_k = \infty. \quad (\text{II.18})$$

- Cas (ii), sous l'hypothèse que la fonction $\nabla \mathcal{J}$ est lipschitzienne sur $\mathcal{X}$, c'est-à-dire qu'il existe $L > 0$ tel que :

$$\forall \mathbf{f}, \mathbf{g} \in \mathcal{X}, \quad \|\nabla \mathcal{J}(\mathbf{f}) - \nabla \mathcal{J}(\mathbf{g})\| \leq L \|\mathbf{f} - \mathbf{g}\|. \quad (\text{II.19})$$

Cette hypothèse impose des conditions sur la suite (s_k) , qui sont moins restrictives que (II.18). En effet, il suffit que :

$$\forall k, 0 < s_k < 2/L \quad (\text{II.20})$$

pour garantir la convergence de l'algorithme du gradient projeté. En fait, la condition (II.20) garantit même la décroissance de la suite $\mathcal{J}(\mathbf{f}^{(k)})$.

II.3.3 Choix retenu

Nous retenons la dernière option, qui laisse le plus de liberté sur le choix de la suite (s_k) . Le caractère lipschitzien de $\nabla \mathcal{J}$ est discuté en annexe A, et démontré dans le cas d'une fonction ϕ de type $\mathcal{L}_2$ - $\mathcal{L}_1$, en particulier pour la fonction de Huber ϕ_T définie en (II.7). En revanche, les fonctions ϕ de type $\mathcal{L}_\gamma$ ne conviennent pas, car elles ne sont pas quadratiques en 0. Pour chaque itération k , nous sélectionnons le paramètre s_k en résolvant la minimisation de ligne :

$$s_k = \arg \min_{s \in (0, 2/L)} \mathcal{J}(\mathbf{f}^{(k)} - s \nabla \mathcal{J}(\mathbf{f}^{(k)})). \quad (\text{II.21})$$

Le schéma présenté par l'équation (II.17) se généralise dans les cas où la direction de descente n'est pas $\nabla \mathcal{J}(\mathbf{f}^{(k)})$. Il est bien connu que les méthodes itératives de descente sont plus performantes en rapidité de convergence en considérant des directions « corrigées » de celle du gradient : correction de la bissectrice, de Vignes. Dans les cas où le minimum $\mathbf{f}^*$ de $\mathcal{J}$ se trouve dans une vallée très étroite, l'algorithme du gradient projeté produit en effet des itérées qui oscillent autour de la position $\mathbf{f}^*$. Il est alors plus efficace de remplacer (II.17) et (II.21) par :

$$\mathbf{f}^{(k+1)} = [\mathbf{f}^{(k)} - s_k \mathbf{a}_k]^+ \quad \text{et} \quad s_k = \arg \min_{s \in (0, 2/L)} \mathcal{J}(\mathbf{f}^{(k)} - s \mathbf{a}_k), \quad (\text{II.22})$$

où $\mathbf{a}_k$ est la direction corrigée. Un choix courant est la direction de la bissectrice entre $\nabla \mathcal{J}(\mathbf{f}^{(k)})$ et la direction de descente à l'itération précédente :

$$\mathbf{a}_k = \|\nabla \mathcal{J}(\mathbf{f}^{(k)})\| \text{ bissec}(\mathbf{a}_{k-1}, \nabla \mathcal{J}(\mathbf{f}^{(k)})), \quad (\text{II.23})$$

où la fonction bissec renvoie un vecteur unitaire. C'est le choix que nous faisons, et l'algorithme utilisé est finalement résumé par la figure II.3 et le tableau II.1.

Réaliser la rétroprojection filtrée $\tilde{\mathbf{f}}^{(0)}$ des données $\mathbf{d}$.
 Initialiser avec $\mathbf{f}^{(0)}$, obtenu en seuillant $\tilde{\mathbf{f}}^{(0)}$ sur $\mathcal{X}$. Choisir $\mathbf{a}_{-1} = 0$.
 Pour $k = 0, \dots, K - 1$,
 Calculer $\nabla \mathcal{J}(\mathbf{f}^{(k)})$.
 Choisir la direction de descente $\mathbf{a}_k$ par (II.23).
 /* Minimisation de ligne */
 Résoudre la minimisation de ligne de (II.22), puis calculer $\tilde{\mathbf{f}}^{(k)} = \mathbf{f}^{(k)} - s_k \mathbf{a}_k$.
 /* Projection sur l'ensemble des contraintes */
 Seuiller $\tilde{\mathbf{f}}^{(k)}$ sur $(0, 1)^n \rightarrow \mathbf{f}^{(k+1)}$.
 Fin Pour.

Tableau II.1 – Algorithme du gradient projeté pour la reconstruction par voxels.

II.3.4 Initialisation

L'initialisation de l'algorithme n'est pas un point clé dans le sens où elle ne conditionne pas le résultat final, le critère $\mathcal{J}(\mathbf{f})$ étant convexe (elle conditionne surtout le coût numérique du calcul). Nous choisissons pour l'initialisation le résultat de la reconstruction par l'algorithme de rétroprojection filtrée (cf. § I.3.1), que l'on seuille sur l'intervalle $(0, 1)^n$. Cette initialisation n'est pas excellente, mais s'obtient de façon rapide.

Dans ce qui suit, nous donnons deux éléments algorithmiques permettant de réduire considérablement le temps de minimisation. Ils consistent à :

- Figurer la valeur de certains voxels lorsque l'on sait qu'ils appartiennent à l'acier ou au défaut, et donc à les extraire du problème de minimisation.
- Améliorer l'initialisation de l'algorithme en utilisant une stratégie multirésolution.

Nous examinons ces deux points dans les sections II.4 et II.5 respectivement.

II.4 Réduction du nombre de voxels

Dans cette partie, nous développons le premier point évoqué ci-dessus : nous réduisons le nombre de paramètres en figeant les valeurs de certains voxels. Le principe est d'accélérer la minimisation de $\mathcal{J}$ en ne conservant qu'une partie des voxels de $\mathbf{f}$. Pour cela, nous utilisons une représentation de type « matrice creuse », c'est-à-dire que nous ne stockons en mémoire que les indices des voxels non nuls et leurs valeurs. Nous pourrions figer les voxels de valeur 1 (voire à la fois les voxels nuls et ceux de valeur 1), mais ceux-ci constituent un volume minoritaire de la scène. De plus, le terme de régularisation $\mathcal{U}(\mathbf{f})$ défini en équation (II.11), favorise ce choix, car il préconise les valeurs nulles des composantes f_i . Notre procédure d'« exclusion » de voxels est discutée au § II.5, et nous nous contentons ici d'adapter la procédure d'estimation des parties II.2 et II.3.

Notation 1 Notons $\mathcal{V} \subset \{1, \dots, n\}$ (resp. $\mathcal{W} = \{1, \dots, n\} - \mathcal{V}$) l'ensemble des voxels non figés, appelés voxels libres ou actifs (resp. ensemble des voxels passifs), v leur nombre et $\mathbf{f}_v$ le sous-vecteur de $\mathbf{f}$ constitué des valeurs de ces voxels.

En notant $\mathbf{A}_v$ la nouvelle matrice de projection, matrice extraite de $\mathbf{A}$ de taille $M \times v$, nous récrivons l'expression (II.11) de l'énergie $\mathcal{J}(\mathbf{f})$ sous la forme :

$$\begin{aligned} \mathcal{J}(\mathbf{f}) &= \mathcal{K}(\mathbf{f}) + \lambda \mathcal{D}(\mathbf{f}) + \mu \mathcal{U}(\mathbf{f}) \\ &= \mathcal{K}_v(\mathbf{f}_v) + \lambda \mathcal{D}_v(\mathbf{f}_v) + \mu \mathcal{U}_v(\mathbf{f}_v) \equiv \mathcal{J}_v(\mathbf{f}_v), \end{aligned} \quad (\text{II.24})$$

où $\mathcal{K}_v(\mathbf{f}_v) = \|\mathbf{d} - \mathbf{A}_v \mathbf{f}_v\|^2$, $\mathcal{U}_v(\mathbf{f}_v) = \sum_{i \in \mathcal{V}} f_i$, et l'opérateur $\mathcal{D}_v$ est défini par :

$$\mathcal{D}_v(\mathbf{f}_v) = \sum_{\substack{i, j \in \mathcal{V} \\ i \sim j}} \phi_T(f_i - f_j) + \sum_{i \in \partial \mathcal{V}} \phi_T(f_i) n_v(i), \quad (\text{II.25})$$

où pour tout voxel $i \in \partial \mathcal{V}$,

$$n_v(i) = \text{Card} \{j \in \mathcal{W}, j \sim i\} \quad (\text{II.26})$$

est le nombre de facettes de i qui sont à la frontière entre $\mathcal{V}$ et $\mathcal{W}$. Ce terme est donc indépendant de $\mathbf{f}$. Contrairement à $\mathcal{U}_v$, opérateur défini sur la base de (II.11), l'énergie $\mathcal{D}_v$ tient compte implicitement de la topologie relative des volumes dénotés par $\mathcal{V}$ et $\mathcal{W}$. Il résulte de (II.24) que le minimiseur global de $\mathcal{J}_v$ sur $(0, 1)^n$ vérifie :

$$\mathbf{f}_v^* \equiv \arg \min_{(0,1)^v} \mathcal{J}_v(\mathbf{f}_v) = \arg \min_{(0,1)^v} \mathcal{J}(\mathbf{f}_v, \mathbf{0}_{n-v}),$$

où $\mathbf{0}_{n-v}$ désigne le vecteur nul de taille $n - v$. En revanche, cela ne signifie pas que $\mathbf{f}_v^*$ est confondu avec la restriction de $\mathbf{f}^*$, minimiseur non contraint de $\mathcal{J}$, à l'ensemble $\mathcal{V}$ des voxels actifs.

L'énergie $\mathcal{J}_v$ est convexe comme somme de fonctionnelles convexes, et son gradient est lipschitzien pour les mêmes raisons que celui de $\mathcal{J}$. Par conséquent, nous pouvons également utiliser un algorithme d'optimisation du premier ordre de type gradient projeté pour estimer $\mathbf{f}_v^*$, la seule différence résidant dans la valeur L_v du facteur de Lipschitz de $\nabla \mathcal{J}_v$ (cf. annexe A).

Le problème est de décider de quelle façon rendre passifs les voxels de $\mathbf{f}$. Nous répondons à cette question dans ce qui suit, avec l'utilisation d'une stratégie multirésolution.

Réaliser la rétroprojection filtrée $\tilde{\mathbf{f}}_1^{(0)}$ des données $\mathbf{d}$. $\tilde{\mathbf{f}}_1^{(0)}$ est de taille $n/8^{R-1}$.
 Seuiller les voxels de $\tilde{\mathbf{f}}_1^{(0)}$ sur l'intervalle $(0, 1) \rightarrow \mathbf{f}_1^{(0)}$.
 Définir $\mathcal{V}_1$, ensemble des voxels non nuls de $\mathbf{f}_1^{(0)}$.
 Pour $r = 1, \dots, R$,
 /* Enlever virtuellement les voxels passifs */
 Si $r \neq 1$, définir $\mathcal{V}_r$ puis $\mathbf{f}_r^{(0)} = \mathcal{I}_r(\hat{\mathbf{f}}_{r-1})$ selon le schéma de la figure II.4.
 /* Ne modifie pas la forme de $\mathcal{S}_{r-1}$ */
 Évaluer $\hat{\mathbf{f}}_r = \arg \min_{(0,1)^{v_r}} \mathcal{J}_r(\mathbf{f}_r)$ à partir de $\mathbf{f}_r^{(0)}$ par l'algorithme monorésolution.
 Fin Pour.

Tableau II.2 – Algorithme multirésolution pour la reconstruction par voxels.

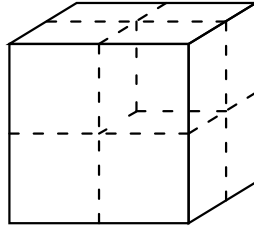


Figure II.4 – Passage d'une résolution à la suivante, par l'exclusion des voxels de valeur nulle, et la division de chaque voxel actif en 8 « sous-voxels » identiques de même densité.

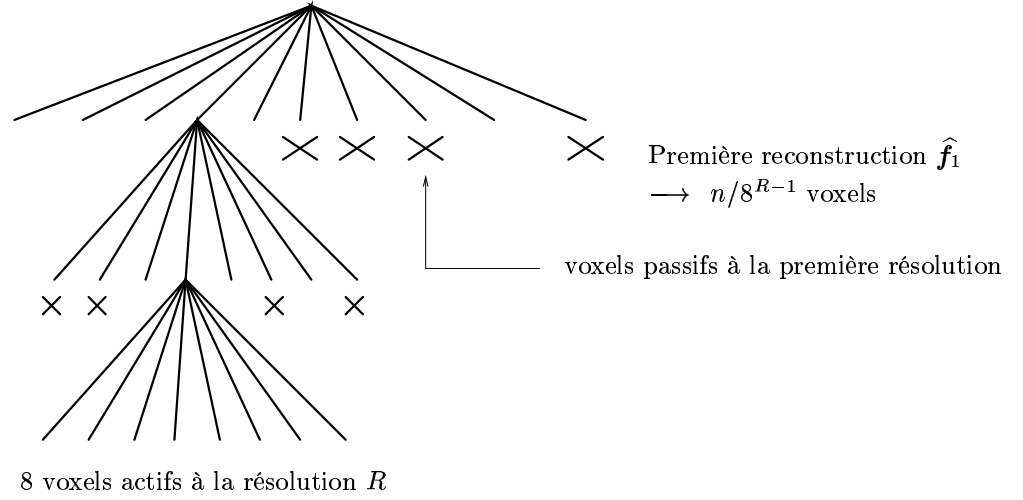


Figure II.5 – Stratégie multirésolution. Concept pyramidal, représentation par structure *octree*.

II.5 Stratégie multirésolution

Afin de mettre en œuvre la réduction du nombre de voxels étudiée précédemment, nous adoptons une stratégie multirésolution au sens où nous discrétisons la scène $\mathcal{S}$ de plus en plus finement, en $\mathbf{f}_1, \dots, \mathbf{f}_R$, en notant R le nombre de résolutions (Bouman et Shapiro, 1994). Les estimations successives $\hat{\mathcal{S}}_r \equiv \hat{\mathbf{f}}_r$ sont effectuées par la méthode de la section II.4 avec pour initialisation les scènes $\hat{\mathbf{f}}_{r-1}$. En particulier, les voxels figés à la résolution r sont les voxels nuls de $\hat{\mathbf{f}}_{r-1}$. Outre la réduction très nette du nombre de voxels actifs au niveau final R , la stratégie multirésolution permet d'obtenir une initialisation $\mathbf{f}_R^{(0)}$ bien meilleure que celle de l'algorithme monorésolution. La R -ème optimisation est donc nettement plus rapide.

II.5.1 Principe

Nous décrivons maintenant cet algorithme de façon plus formelle. Tout d'abord, nous introduisons la transformation $\mathcal{I}_r$, qui associe à l'image $\hat{\mathbf{f}}_{r-1}$ son image « fille » $\mathbf{f}_r^{(0)}$ à la résolution r :

- chaque voxel actif de $\hat{\mathbf{f}}_{r-1}$ et de valeur nulle est rendu passif.
- les autres voxels actifs de $\hat{\mathbf{f}}_{r-1}$ sont divisés en 8 sous-voxels ou *files* (actifs) de même densité.

Cette transformation, illustrée par la figure II.4, ne modifie donc pas la nature de la scène $\hat{\mathbf{f}}_{r-1}$. En supposant que n soit le nombre final de voxels (voxels actifs à la résolution R et descendances « virtuelles » des voxels passifs) et que n soit une puissance de 8, nous commençons la reconstruction au niveau $r = 1$ avec $n/8^{R-1}$ voxels.

Notation 2 À chaque niveau r de résolution, nous définissons :

- La fonctionnelle $\mathcal{J}_r = \mathcal{J}_{\mathcal{V}_r}$ selon (II.24). $\mathcal{V}_r$ est un sous-ensemble de $\{1, \dots, n/8^{R-r}\}$. Le nombre de voxels actifs est $v_r = \text{Card}(\mathcal{V}_r)$, et le vecteur image $\mathbf{f}_{v_r}$ est renommé $\mathbf{f}_r$.
- La transformation $\mathcal{I}_r : \mathbb{R}^{v_{r-1}} \rightarrow \mathbb{R}^{v_r}$, qui associe à toute image à la résolution $r-1$ dont le support est inclus dans celui de $\hat{\mathbf{f}}_{r-1}$, son image « fille » à la résolution r .
- Enfin, si le voxel i est le père du voxel k , nous notons $k \in i$, et $i = \uparrow k$.

L'algorithme décrit par le tableau II.2 réalise les minimisations successives des fonctions $\mathcal{J}_r(\mathbf{f}_r)$ à partir de $\mathbf{f}_r^{(0)} = \mathcal{I}_r(\widehat{\mathbf{f}}_{r-1})$ sur le domaine $(0, 1)^{v_r}$, par l'algorithme monorésolution décrit par le tableau II.1.

Le schéma de multirésolution s'interprète donc en un arbre dont chaque nœud correspond à un voxel actif et possède 8 fils (structure pyramidale de type *quadtree* en 2D et *octree* ici (Bouman et Shapiro, 1994; Pérez et Heitz, 1996)). La profondeur de l'arbre est le nombre R de niveaux de résolution et ses feuilles sont les voxels passifs aux résolution $r = 1, \dots, R-1$, et les voxels actifs à la résolution R (cf. figure II.5).

II.5.2 Réglage des hyperparamètres

Nous précisons ici le choix des hyperparamètres λ_r , μ_r et T_r à chaque niveau de résolution. Ces choix sont faits de manière récursive ; en effet, nous montrons qu'il existe des conditions suffisantes sur ces suites de paramètres qui assurent la condition :

$$\mathcal{J}_{r-1}(\widehat{\mathbf{f}}_{r-1}) = \mathcal{J}_r(\mathbf{f}_r^{(0)}), \quad (\text{II.27})$$

et garantissent finalement :

$$\forall r \in \{1, \dots, R\}, \forall k \in \{1, \dots, r-1\}, \quad \mathcal{J}_r(\widehat{\mathbf{f}}_r) \leq \mathcal{J}_{r-k}(\widehat{\mathbf{f}}_{r-k}). \quad (\text{II.28})$$

Pour garantir la condition (II.27), nous choisissons de fixer les paramètres T_r à une valeur constante T . Ce choix permet de pénaliser de la même façon deux voxels adjacents à la résolution r , dont les pères sont adjacents à la résolution $r-1$, compte tenu du fait que $\mathbf{f}_{rk} = \mathbf{f}_{r-1, \uparrow k}$, $\forall k$. Pour satisfaire (II.27), nous obtenons des conditions suffisantes sur λ_r et μ_r de façon à satisfaire $\lambda_{r-1} \mathcal{D}_{r-1}(\widehat{\mathbf{f}}_{r-1}) = \lambda_r \mathcal{D}_r(\mathbf{f}_r^{(0)})$, puis $\mu_{r-1} \mathcal{U}_{r-1}(\widehat{\mathbf{f}}_{r-1}) = \mu_r \mathcal{U}_r(\mathbf{f}_r^{(0)})$.

Proposition 1 Les énergies $\mathcal{D}_{r-1}$ et $\mathcal{D}_r$ vérifient $\mathcal{D}_{r-1}(\widehat{\mathbf{f}}_{r-1}) = 4 \mathcal{D}_r(\mathbf{f}_r^{(0)})$.

Proposition 2 Les énergies $\mathcal{U}_{r-1}$ et $\mathcal{U}_r$ vérifient $\mathcal{U}_{r-1}(\widehat{\mathbf{f}}_{r-1}) = 8 \mathcal{U}_r(\mathbf{f}_r^{(0)})$.

Ces deux propositions sont prouvées dans l'annexe II.A. Nous en déduisons une condition suffisante sur les hyperparamètres pour la réalisation de (II.27) :

$$\lambda_r = \lambda_{r-1}/4, \quad (\text{II.29})$$

$$\mu_r = \mu_{r-1}/8, \quad (\text{II.30})$$

$$T_r = T. \quad (\text{II.31})$$

En particulier, nous avons

$$\lambda_r = \lambda^0/4^R, \quad \mu_r = \mu^0/8^R. \quad (\text{II.32})$$

En pratique, nous réglons uniquement les deux paramètres finaux λ_R et μ_R , et le seuil T .

II.6 Conclusions

Dans ce chapitre, nous avons étudié les méthodes de reconstruction par voxels d'une scène 3D à partir d'un faible nombre de vues. Nous avons ramené ce problème à l'estimation des valeurs des voxels au sens du MAP, c'est-à-dire à la minimisation de l'énergie *a posteriori* $\mathcal{J}(\mathbf{f})$ sur l'intervalle $(0, 1)^n$. Nous avons choisi par simplicité une énergie convexe, dont l'optimisation s'effectue par l'algorithme itératif du gradient projeté. Cet algorithme s'avère extrêmement lent pour les grandes valeurs de n ($n \geq 10^9$). Par conséquent, nous avons été amenés à proposer des stratégies originales pour améliorer ses performances en termes de coût algorithmique. Nous les illustrerons par ailleurs par les simulations du chapitre VI.

II.A Preuves des propositions 1 et 2

Preuve 1 Nous récrivons les expressions des fonctionnelles $\mathcal{D}_k$, définies à partir de (II.25), sous la forme :

$$\mathcal{D}_k(\mathbf{f}_k) = \sum_{\substack{i, j=1 \\ i \sim j}}^{n/8^{R-k}} \phi_T(f_k(i) - f_k(j)), \quad (\text{II.33})$$

où $f_k : \{1, \dots, n/8^{R-k}\} \longrightarrow (0, 1)$ est la fonction définie par :

$$f_k(i) = \begin{cases} f_{ki} & \text{si } i \in \mathcal{V}_k, \\ 0 & \text{sinon.} \end{cases}$$

Considérons à présent l'image $\mathbf{f}_r^{(0)}$ et un couple (i, j) de voxels adjacents dans $\mathcal{V}_r$. Deux situations peuvent se produire :

- i et j ont le même père, alors $f_r^{(0)}(i) = f_r^{(0)}(j) = \widehat{f}_{r-1}(\uparrow i)$. Comme $\phi_T(0) = 0$, nous avons $\phi_T(f_r^{(0)}(i) - f_r^{(0)}(j)) = 0$.
- i et j n'ont pas le même père, leur pères sont donc des voxels adjacents de $\mathcal{V}_{r-1}$. Nous avons finalement $\phi_T(f_r^{(0)}(i) - f_r^{(0)}(j)) = \phi_T(\widehat{f}_{r-1}(\uparrow i) - \widehat{f}_{r-1}(\uparrow j))$.

Finalement, nous avons :

$$\begin{aligned} \mathcal{D}_r(\mathbf{f}_r^{(0)}) &= \sum_{\substack{i, j=1 \\ i \sim j, \uparrow i \neq \uparrow j}}^{n/8^{R-r}} \phi_T(\widehat{f}_{r-1}(\uparrow i) - \widehat{f}_{r-1}(\uparrow j)) \\ &= 4 \sum_{\substack{i, j=1 \\ i \sim j}}^{n/8^{R-r+1}} \phi_T(\widehat{f}_{r-1}(i) - \widehat{f}_{r-1}(j)) = 4 \mathcal{D}_r(\widehat{\mathbf{f}}_{r-1}), \end{aligned}$$

car chaque couple de voxels adjacents à la résolution $r-1$ génère exactement 4 sous-couples de voxels adjacents à la résolution r . $\square$

Preuve 2 Comme chaque voxel actif $\widehat{f}_{r-1,i}$ se décompose en 8 sous-voxels de même densité, et les voxels passifs ont une valeur nulle, nous avons :

$$\sum_{i \in \mathcal{V}_{r-1}} \widehat{f}_{r-1,i} = 8 \sum_{i \in \mathcal{V}_r} f_{r,i}^{(0)}.$$

□

CHAPITRE III

APPROCHE CONTOUR

- III.1 Introduction
- III.2 Modélisations de contours
- III.3 Sélection de modèles adéquats
- III.4 Étude des modèles retenus
- III.5 Conclusions

III.1 Introduction

Dans ce chapitre, nous considérons spécifiquement les modèles de contours déformables, aussi appelés « modèles géométriques », pour la reconstruction tomographique d'un objet compact $\mathcal{C}^*$ à partir d'un faible nombre de vues. Bien que nos motivations concernent la reconstruction de formes 3D, l'exposé des modèles 2D est souvent plus simple (dans le cadre cette fois de la reconstruction d'objets plans en tomographie bidimensionnelle, que nous avons décrite au chapitre I). Pour la plupart des modèles, nous exposons donc au préalable le modèle 2D correspondant, puis l'étendons au cas 3D, en soulignant les difficultés théoriques et pratiques, et les techniques nécessaires à cette extension.

Avant de poursuivre, nous rappelons que l'objet réel est défini par l'intérieur de son contour $\mathcal{C}^*$. La surface déformable¹ est notée $\mathcal{C}$, et la fonction de densité associée est :

$$f(x, y, z) = \begin{cases} 1 & \text{si } M(x, y, z) \in \mathcal{C}, \\ 0 & \text{sinon.} \end{cases} \quad (\text{III.1})$$

Ses projections *ponctuelles*, décrites par les équations (I.10) et (I.14), se récrivent simplement :

$$p_{\theta}(\mathbf{t}) = \int_{\mathcal{L}_{\mathbf{t}, \theta}} f(x, y, z) dl = \int_{\mathcal{L}_{\mathbf{t}, \theta} \cap \mathcal{C}} dl. \quad (\text{III.2})$$

En d'autres termes, chaque valeur $p_{\theta}(\mathbf{t})$ d'une projection est directement liée aux intersections du rayon correspondant $(\mathcal{L}_{\mathbf{t}, \theta})$ avec la surface $\mathcal{C}$. Le problème de la reconstruction de $\mathcal{C}^*$ à partir d'un faible nombre de ses projections bruitées (les projections de $\mathcal{C}^*$ sont « non tronquées » au sens où le champ de vue englobe leur support) se ramène finalement à l'estimation du contour déformable $\mathcal{C}$ à partir des données.

Le présent chapitre est composé de trois parties. Au § III.2, nous proposons une synthèse des modèles géométriques en imagerie (et notamment en tomographie), en distinguant en particulier les contours actifs des contours paramétriques. Nous concluons cette étude au § III.3 en retenant des modèles géométriques adéquats pour notre problème. En outre, nous proposons une structure d'estimation de leurs paramètres à partir des données (estimation au sens des moindres carrés contraints ou non, ou du maximum *a posteriori*). Enfin, le § III.4 détaille les modèles sélectionnés en précisant leurs intérêts, leurs limitations, et pour certains nos contributions.

¹Par simplicité, nous désignons par $\mathcal{C}$ à la fois l'intérieur de l'objet et son contour.

III.2 Modélisations de contours

Dans la littérature du traitement d'images, et plus particulièrement en segmentation d'images et en vision par ordinateur, les modèles de contours déformables ont été largement utilisés. Nous pouvons distinguer deux classes de modèles déformables :

1. Les modèles non paramétriques qui, sous leur forme explicite, définissent le contour $\mathcal{C}$ à partir d'une abscisse curviligne s en 2D, de deux abscisses curvilignes $\mathbf{s} = (s_1, s_2)$ en 3D. $\mathcal{C}$ est alors décrit par une fonction $u(\mathbf{s})$ à valeurs dans $\mathbb{R}^2$ ou $\mathbb{R}^3$, où u est régulière (de classe $\mathcal{C}^2$) et périodique en $\mathbf{s}$ de façon à générer un contour fermé. Ces modèles, appelés *contours actifs* ou *snakes*, sont peu restrictifs sur la forme de $\mathcal{C}$, mais imposent de gérer le déplacement du contour en chaque point $u(\mathbf{s})$.
2. Les modèles *paramétriques*, notés $\mathcal{C}_{\mathbf{x}}$, qui dépendent d'un nombre fini de paramètres $\mathbf{x}$, et caractérisent des formes plus ou moins spécifiques. Le problème de reconstruction du contour $\mathcal{C}_{\mathbf{x}}$ se ramène donc à l'estimation des paramètres $\mathbf{x}$ directement à partir des données.

Nous développons ces modèles dans les deux sections suivantes, en mettant l'accent sur leur intérêt pour le problème de la reconstruction tomographique.

III.2.1 Modélisations non paramétriques

Le but de cette partie n'est pas de réaliser une synthèse exhaustive de ces modèles, aussi appelés contours actifs, mais de les introduire en insistant sur leurs intérêts et leurs limites pour les problèmes de reconstruction.

III.2.1.1 Principe des contours actifs

Le modèle des contours actifs ou *snakes* a été introduit en vision par ordinateur et en particulier en segmentation d'images 2D par Kass et al. (1988). L'idée est de définir le contour $\mathcal{C}$ par une fonction de classe $\mathcal{C}^2$, notée $u : (0, 1) \rightarrow \mathbb{R}^2$, qui minimise une énergie

$$\mathcal{J}(u) = \mathcal{K}(u) + \mathcal{R}(u), \quad (\text{III.3})$$

où $\mathcal{K}$ est une énergie externe (liée aux données), et $\mathcal{R}$ est une énergie interne, elle-même composée de deux termes :

$$\mathcal{R}(u) = \lambda_1 \int_0^1 \|u'(t)\|^2 dt + \lambda_2 \int_0^1 \|u''(t)\|^2 dt. \quad (\text{III.4})$$

Ce modèle s'inspire à la base de la mécanique des solides, où l'objet $\mathcal{C}$ est conçu pour se déformer sous l'effet de deux forces liées respectivement aux données et à des potentiels qui contraignent sa régularité. Physiquement, les termes (III.4) qui constituent l'énergie interne s'interprètent comme l'élasticité (liée à la longueur de $\mathcal{C}$) et la rigidité (liée à sa courbure, elle pénalise ses torsions) du contour. Nous renvoyons le lecteur au § IV.2.2.2 pour de plus amples développements sur ces termes. En termes de traitement de données, l'énergie externe $\mathcal{K}(u)$ traduit l'attache aux données, alors que $\mathcal{R}(u)$ représente l'*a priori* de régularité du contour $\mathcal{C}$. Par exemple, en segmentation d'images, $\mathcal{K}(u)$ est liée à l'image $\mathbf{I}$ à segmenter, ou plutôt à son gradient :

$$\mathcal{K}(u) = - \int_0^1 |\nabla \mathbf{I}(u(t))|^2 dt. \quad (\text{III.5})$$

III.2.1.2 Mise en œuvre des contours actifs

L'application des contours actifs fait traditionnellement appel aux techniques variationnelles pour le calcul de la solution. On montre que le minimiseur $\hat{u}$ de l'énergie $\mathcal{J}(u)$ satisfait une équation aux dérivées partielles, dont la solution est obtenue d'une manière itérative par des déformations successives d'une solution initiale.

Bien que les snakes modélisent des formes aussi variées que possibles, ils sont surtout adaptés à des données peu bruitées. De plus, ils soulèvent des problèmes majeurs : la dépendance de la solution par rapport à la paramétrisation choisie pour u , et le coût de calcul de la solution, dû en grande partie à la mise à jour du contour sur sa globalité (en tout point $u(t)$).

III.2.1.3 Modèles implicites

Pour pallier le principal défaut du modèle précédent, *i.e.*, le fait qu'il ne gère pas les changements de topologie du contour, Osher et Sethian (1988) ont proposé de définir un contour actif 2D sous une forme implicite, comme la courbe de niveau 0 d'une fonction Φ définie sur l'ensemble de la scène :

$$\mathcal{C} = \{M(x, y) \in \mathbb{R}^2, \Phi(x, y) = 0\}, \text{ où } \Phi : \mathbb{R}^2 \rightarrow \mathbb{R}. \quad (\text{III.6})$$

Plus précisément, l'intérieur de $\mathcal{C}$ est l'ensemble des points en lesquels Φ est négative. Le principe de l'estimation de $\mathcal{C}$ est le suivant. Il s'agit successivement de :

1. définir une loi d'évolution ($\Phi(x, y, \tau)$ dépend à présent d'une variable « temporelle » τ) sous la forme $\partial\Phi(x, y, \tau)/\partial\tau + F \|\nabla\Phi(x, y, \tau)\| = 0$, où F correspond à la vitesse d'évolution du contour, et dépend des données et de la courbure de $\mathcal{C}_\tau$ (Malladi et al., 1995) ;
2. puis de mettre à jour alternativement la scène $\Phi(x, y, \tau)$ au temps τ , et sa courbe de niveau $\mathcal{C}_\tau$.

Dans le domaine de la reconstruction tomographique, on procède traditionnellement en deux étapes : la reconstruction tomographique d'une carte de densité non binaire (méthode par voxels), puis sa segmentation par contours actifs (Teboul et al., 1997). Néanmoins, depuis quelques années, des chercheurs ont généralisé le modèle par courbe de niveau à la reconstruction directe de contour 2D en tomographie (Santosa, 1996; Litman et al., 1997; Whitaker, 1998). Nous ne retenons pas cette approche, car elle nécessite une transformation de la scène en voxels pour deux raisons :

- le calcul de $\Phi(x, y, \tau)$ sur tout le volume de la scène,
- l'évaluation des projections du contour.

En d'autres termes, il s'agit d'une approche « mixte » (à la fois paramétrique et par voxels), bien que le modèle soit géométrique.

III.2.1.4 Conclusion

Au vu des développements précédents, nous n'optons pas pour les contours actifs définis sous leur forme générale (paramétrisation continue) pour les raisons suivantes :

- La projection d'un contour actif ne peut être évaluée qu'approximativement, soit en « voxellisant » l'intérieur du contour (voir § IV.4.1 pour la description de ce procédé pour un cas particulier de contour), soit en approchant le contour par une spline. De plus, ces approximations s'avèrent complexes car nous n'avons pas d'information sur la forme de $\mathcal{C}$.
- La régularité de la fonction u est un obstacle à la détection de singularités dans le contour.

Nous préférons en outre réduire le nombre d'inconnues en optant pour un modèle *paramétrique*. En particulier, nous recherchons des modèles qui réalisent un bon compromis entre le nombre de paramètres et la diversité de formes engendrées.

III.2.2 Modélisations paramétriques

Nous décrivons maintenant (de façon non exhaustive) un ensemble de modèles paramétriques, en allant des plus particuliers aux plus complexes (au sens où les hypothèses *a priori* sur la forme sont les plus lâches). Ces modèles sont classés en deux catégories suivant le nombre n de leur paramètres. Nous étudions d'une part les modèles à très faible nombre de paramètres (par exemple les ellipsoïdes), d'autre part les modèles à grand nombre de paramètres (n est alors généralement relié à la résolution spatiale du contour). Nous attirons l'attention du lecteur sur le fait que l'un des problèmes majeurs posé par l'application tomographique est le calcul de la projection du contour. Pour la clarté de l'exposé, nous laissons ce problème en suspens, et le traitons au § III.3.

III.2.2.1 Modèles à faible nombre de paramètres

Ces modèles représentent des objets géométriques très simples. Ils interviennent dans les situations où nous avons une information précise sur la forme à reconstruire, comme pour les organes en imagerie médicale. Nous distinguons deux représentations de ces modèles.

Modèles « directs ». Ils caractérisent directement les propriétés physiques du contour : aire ou volume intérieur, centre et axes d'inertie, points de contrôle du contour. L'une des contributions pionnières pour leur reconstruction à partir de données tomographiques est celle de Rossi et Willsky (1984). Ils reconstruisent des scènes constituées d'objets compacts paramétriques et uniformes dans un fond connu, en mettant l'accent sur la qualité de la reconstruction de paramètres d'intérêt, plus que la qualité de l'image reconstruite. Ils traitent en particulier le cas des formes elliptiques, l'un des modèles de contours fermés les plus simples, car ces formes sont totalement décrites par leurs moments d'inertie d'ordre inférieur ou égal à 2.

Par la suite, ces derniers modèles ont été largement utilisés en tomographie, notamment pour des applications médicales (reconstruction d'arbres artériels en angiographie, les ellipses modélisant les coupes d'artères de façon très appropriée (Bresler et Macovski, 1987; Fessler et Macovski, 1991)), mais aussi en vision par ordinateur (reconstruction à partir de silhouettes (Prince et Willsky, 1991)). L'extension de ce modèle en 3D est le modèle ellipsoïdal, dont la paramétrisation et la reconstruction à partir de silhouettes ont été traitées par Karl et al. (1994). D'une façon plus générale, les superquadriques et les hyperquadriques dépendent aussi d'un faible nombre de paramètres, mais produisent des formes non convexes, bien qu'elles possèdent des axes de symétrie (Solina et Bajcsy, 1990; Bardin et al., 1994). Nous reviendrons sur ces formes simples au § III.4.1.

Il existe également des situations où l'objet est partiellement connu, donc naturellement paramétrable. Par exemple, Lagrange (1998) étudie la reconstruction tomographique d'une scène 2D homogène par morceaux et à symétrie circulaire. Pour cela, il définit une fonction de densité 1D affine par morceaux, qu'il paramètre par des informations physiques relatives à la scène (positions des points de transition entre matériaux, valeurs de la densité en ces points).

Modèles « indirects ». Ce sont les modèles de transformation d'une forme géométrique simple de référence, notée $\mathcal{C}_{\text{REF}}$. Il s'agit cette fois de définir une classe d'applications régulières $\Phi_{\mathbf{x}} : \mathcal{C}_{\text{REF}} \rightarrow \mathcal{C}_{\mathbf{x}}$, décrites par des paramètres $\mathbf{x}$. Choisir une forme $\mathcal{C}_{\mathbf{x}}$, c'est donc choisir le jeu de paramètres $\mathbf{x}$ d'une telle transformation. Cette approche présente l'avantage de préserver les caractéristiques topologiques de $\mathcal{C}_{\text{REF}}$, et donc d'éviter les problèmes d'auto-intersections du contour $\mathcal{C}_{\mathbf{x}}$. Elle a été notamment utilisée par Amit et Manbeck (1993) pour la reconstruction de tumeurs au cerveau en tomographie d'émission. Bien que leur modèle ne génère pas une densité uniforme par morceaux (elle est en fait homogène par morceaux), il entre bien dans cette classe de modèles : l'objet de référence $\mathcal{C}_{\text{REF}}$ est une ellipse de densité uniforme, et les transformations $\Phi_{\mathbf{x}}$ portent à la fois sur le contour elliptique et sur sa densité (elles font intervenir deux transformations affines). Milanfar et al. (1994) utilisent également des transformations affines $\Phi_{\mathbf{x}}$, avec un objet de référence polygonal (polygone régulier,

de densité uniforme), qui mène à la classe des polygones affinements réguliers². Ce modèle est décrit plus en détail à la fin du § IV.2.1.

Les modèles dits de *déformation à forme libre* sont proches des précédents, et entrent aussi dans la catégorie des modèles « indirects ». En fixant une forme de référence $\mathcal{C}_{\text{REF}}$, le principe est d'inclure $\mathcal{C}_{\text{REF}}$ dans une « boîte » cubique, que l'on rend déformable. Les déformations sont gérées par les déplacements de points de contrôle P_i de la boîte, en associant à ces déplacements une déformation continue de l'espace. Ce modèle a été appliqué par Bardin et al. (1994) en segmentation d'images. Ces auteurs considèrent une superquadrique pour l'objet de référence (préalablement estimée par moindres carrés), et discrétisent au départ la surface du plus petit parallélépipède contenant $\mathcal{C}_{\text{REF}}$ pour initialiser les points P_i . La déformation de l'espace suite aux déplacements des points P_i est gérée en calculant la surface polynomiale d'interpolation des points P_i . Battle et al. (2000) utilisent un modèle analogue pour la reconstruction tomographique de poumons, de tissus humains, et de l'épine dorsale en imagerie médicale nucléaire.

Conclusion. Pour notre problème, nous retenons des modèles directs à faible nombre de paramètres (ellipsoïdes et superquadriques) afin de fournir des reconstructions grossières de l'objet, et d'estimer des paramètres d'intérêt (volume, axes, et moments d'inertie). Nous utiliserons ultérieurement ces reconstructions comme l'initialisation d'une méthode plus élaborée.

III.2.2.2 Modèles à grand nombre de paramètres

Nous abordons ici les modèles à nombre quelconque de paramètres. Ils gèrent des formes plus variées, et le nombre de paramètres est souvent relié à la fréquence spatiale des contours. Parmi ces modèles, nous distinguons les modèles *globaux* des modèles *locaux*, où la notion de localité est liée à la portion du contour que contrôle chaque paramètre (pour un modèle global, une modification de l'un des paramètres engendre un déplacement de toute la surface). Nous illustrons cette notion aux paragraphes suivants par deux classes de contours : les *contours harmoniques* et les *splines*.

Modèles globaux. Sous leur forme la plus simple, les premiers sont obtenus à partir d'une paramétrisation continue $M(t, u) : (0, 2\pi)^2 \rightarrow \mathbb{R}^3$. Cette fonction est notamment supposée de classe $\mathcal{C}^2$ (contour régulier), et vérifie les conditions aux limites $M(t, 0) = M(t, 2\pi)$, $M(0, u) = M(2\pi, u)$ de façon à assurer la fermeture du contour³. Par conséquent, elle admet un développement en série de Fourier. Prenons le cas 2D, traité par Staib et Duncan (1992) pour la segmentation d'images en résonance magnétique nucléaire. Ils utilisent la troncature de ce développement à un ordre L arbitraire :

$$M(t) = (x(t), y(t))^t = G + \sum_{l=1}^L \mathbf{A}_l (\cos lt, \sin lt)^t, \quad (\text{III.7})$$

où les matrices de transformation $\mathbf{A}_l$ sont de taille 2×2 , et le point $G \in \mathbb{R}^2$ est le centre du contour. C'est une approche « multirésolution », au sens où les premiers indices de ce développement correspondent aux basses fréquences du contour. Le modèle (III.7) est aussi appelé « décomposition de Fourier elliptique » de la frontière, car chaque contour fermé d'équation $M(t) = \mathbf{A}_l (\cos lt, \sin lt)^t$ correspond à une ellipse centrée particulière. En d'autres termes, le contour est constitué d'une superposition d'ellipses, comme illustré sur la figure III.1. Staib et Duncan (1996) ont étendu ce modèle au cas 3D avec la décomposition de $M(t, u)$ sur la famille formée du produit tensoriel de $\{\cos kt, \sin kt, k = 1, \dots, L\}$ et $\{\cos lu, \sin lu, l = 1, \dots, L\}$.

Des modèles du même type sont apparus en vision d'objets 3D. Par exemple, Lefaix et al. (1997) utilisent une modélisation harmonique de vertèbres pour le diagnostic de scolioses, basée sur une paramétrisation du contour en coordonnées sphériques : $M(\boldsymbol{\theta}) = G + \rho(\boldsymbol{\theta}) \mathbf{w}_{\boldsymbol{\theta}}$, où les angles $\boldsymbol{\theta} = (\theta, \varphi)$

²Cette fois, les applications $\Phi_{\mathbf{a}}$ portent exclusivement sur le contour.

³De plus, le caractère injectif de $M(t, u)$ garantit l'absence d'auto-intersection.

sont définis sur la figure I.4 de la page 18. Lédée et al. (1997) utilisent un modèle proche pour la restitution du ventricule gauche du cœur en écho-cardiographie. C'est un modèle temporel (donc 4D) de la forme $M(\theta, t) = G + \rho(\theta, t) \mathbf{w}_\theta$. À θ fixé, ils optent pour un développement en série de Fourier de $(\varphi, t) \rightarrow \rho(\theta, \varphi, t)$, car cette fonction est périodique en φ et t , la période en t représentant la durée d'un cycle cardiaque. Les harmoniques sphériques trouvent leur intérêt dans le fait que l'on peut imposer simplement l'injectivité de $M(\theta)$, *i.e.*, le fait que la surface ne s'auto-intersecte pas, par une condition du type $\rho(\theta) > 0, \forall \theta$. Cet argument motive l'utilisation de ce modèle en vue d'une application en reconstruction tomographique. Nous y reviendrons de façon plus précise au § III.4.2.

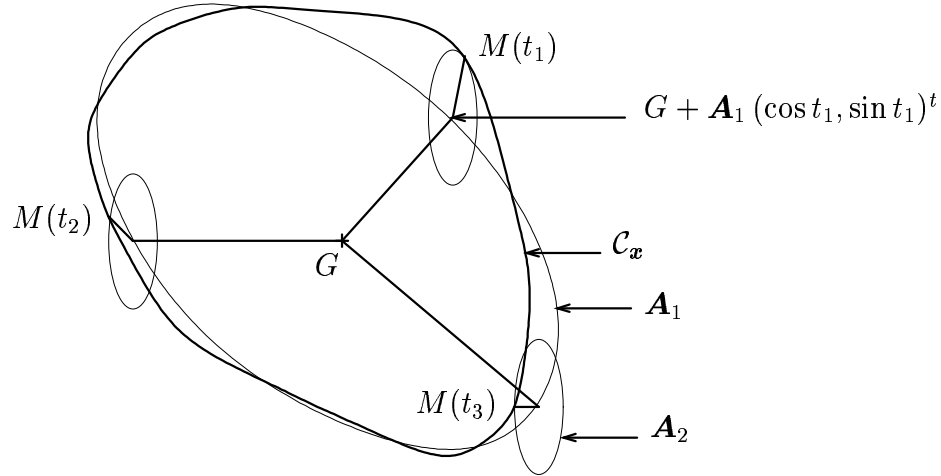


Figure III.1 – Superposition d'ellipses pour le modèle harmonique (III.7), cas où $L = 2$. L'ellipse repérée par $\mathbf{A}_1$ représente les basses fréquences du contour, et la seconde ellipse définit des déformations locales de la première.

Modèles locaux. Nous introduisons maintenant les modèles *locaux*, dont chaque paramètre ne contrôle qu'une portion du contour. C'est en particulier le cas des surfaces splines et des surfaces de Bézier. En reconstruction tomographique, on fixe généralement l'ordre de la forme paramétrique, et on estime les positions de ses points de contrôle à partir des données. Ces modèles permettent d'adapter la distribution des points de contrôle, en les concentrant dans les zones « haute fréquence » des contours. En tomographie, les splines 2D ont été notamment utilisées par Chalmond (1992) et Hanson et al. (1994) en imagerie médicale, et les courbes de Bézier par Cunningham et al. (1996). Senasli et al. (2000) préconise aussi le modèle spline 2D pour la reconstruction coupe par coupe (ou 2,5D) de vaisseaux en angiographie : dans chaque plan de coupe, la section d'un vaisseau est modélisée par une spline dont les points de contrôle sont estimés. Un modèle analogue est retenu par Coutand et al. (1997) pour la reconstruction 2,5D du ventricule gauche du cœur en tomographie SPECT (leur modèle est en fait « mixte », *i.e.*, ils modélisent le contour du ventricule par une spline, et son intérieur (non homogène) par un ensemble de pixels, puis réalisent les estimations successives de la spline et de son intérieur). Enfin, Chalmond et al. (1994) prônent une approche analogue pour la reconstruction tomographique de courbes planes 3D en contrôle non destructif. Quant aux surfaces splines fermées, elles apparaissent principalement sous forme de polyèdres (splines d'ordre 1) pour des raisons algorithmiques. Nous renvoyons le lecteur aux chapitres IV, V et au § III.4.3 pour de plus amples précisions.

III.3 Sélection de modèles adéquats

Nous dégageons à présent de ce qui précède les modèles paramétriques qui nous semblent le plus adéquats à la reconstruction tomographique. Leur sélection est guidée par les critères suivants :

- Les modèles doivent engendrer des formes fermées.
- Ils doivent réaliser un bon compromis entre la variété des formes engendrées et le nombre de paramètres.
- L'évaluation exacte ou approchée de leurs projections doit être simple et directe.

Nous envisageons successivement des formes géométriques simples (§ III.3.1), puis des formes plus variées obtenues par des modèles globaux et locaux (§ III.3.2). Tous ces modèles seront étudiés de façon plus poussée au § III.4. Enfin, le § III.3.3 présente formellement l'estimation des paramètres $\mathbf{x}$ à partir des données, en soulignant les difficultés algorithmiques qui en découlent.

III.3.1 Modèles à faible nombre de paramètres

Parmi les formes décrites au § III.2.2.1, nous optons tout d'abord pour les ellipsoïdes pour leur simplicité. En effet, ils ne font intervenir qu'un faible nombre de paramètres qui décrivent les principales caractéristiques de l'objet. De plus, nous verrons au § III.4.1 que leur estimation (estimation de leurs moments d'inertie) est rendue très simple par la méthode des moments. Par la suite, nous envisageons l'extension de ce modèle aux super-ellipsoïdes, formes également très simples mais *non convexes* en général. En revanche, ces formes possèdent plusieurs plans de symétrie. Elles ne sont donc pas satisfaisantes dans l'optique de la reconstruction de formes complexes. Pour cela, les modèles à grand nombre de paramètres s'imposent.

III.3.2 Modèles à grand nombre de paramètres

Avant de sélectionner des modèles proprement 3D, nous présentons une synthèse des modèles 3D définis coupes par coupes, que nous appelons « 2,5D » (il s'agit d'un assemblage de modèles 2D). En effet, ces modèles sont largement utilisés en tomographie de par leur simplicité.

Modèles « 2,5D ». Ils sont souvent préférés aux modèles 3D dans des situations où l'objet est orienté suivant un axe particulier (pas forcément un axe de symétrie) et les directions de projections sont orthogonales à cet axe. En nous appuyant sur la figure III.2, nous définissons un *cylindre généralisé* (CG) par une courbe axiale $\{\mathbf{N}(t), t \in (0, 1)\}$ de $\mathbb{R}^3$, et des coupes transverses $\mathcal{C}(t)$ définies suivant les plans orthogonaux à cet axe. Dans de nombreuses applications en tomographie (Cinquin, 1981; Bresler et Macovski, 1987; Fessler et Macovski, 1991; Lagrange, 1998; Senasli et al., 2000), on considère la sous-classe des CG engendrée à partir d'un axe linéique, par exemple l'axe vertical $\mathbf{k}$. L'objet 3D est donc défini par un ensemble de coupes $\mathcal{C}(z_0)$, $z_0 \in (0, 1)$, qui se trouvent dans les plans ($z = z_0$) (voir la figure III.2 (b)). En suivant les développements de l'ouvrage (Chalmond, 2000, § 12.3), ce dernier modèle est paramétrable par

$$M(z, u) = g(z, u) \cos u \mathbf{i} + g(z, u) \sin u \mathbf{j} + z \mathbf{k}, \quad (\text{III.8})$$

où $g(z, u)$ est une fonction continue sur $(0, 1) \times \mathbb{R}$ et périodique en u de période 2π (fonction 2D paramétrique). *L'intérêt d'un tel modèle pour la reconstruction tomographique s'exprime dans les cas où les directions de projections $\mathbf{w}_i$ sont toutes orthogonales à $\mathbf{k}$.* En effet, pour une projection fixée, la mesure du détecteur de centre (x_d, y_d, z_d) correspond alors à un rayon (en fait un faisceau) qui se trouve dans le plan ($z = z_d$). Par conséquent, cette mesure fait uniquement intervenir la tranche $\mathcal{C}(z_d) = \{\mathcal{C}(z_d, u), u \in (0, 2\pi)\}$ du modèle qui se trouve dans ce plan. Le problème de la reconstruction de $\mathcal{C} = \cup \{\mathcal{C}(z_d), d \in (0, 1)\}$ est donc équivalent à autant de problèmes de reconstruction 2D que l'on définit de coupes. En tomographie, ces modèles ont été notamment utilisés en détonique (Lagrange, 1998, avec un modèle paramétrique mixte qui porte à la fois sur le contour et la fonction de densité), et pour

la reconstruction d'arbres artériels en angiographie (Bresler et Macovski, 1987; Fessler et Macovski, 1991), avec des sections $\mathcal{C}(z_d)$ elliptiques. Dans toutes ces contributions, les auteurs ne traitent pas exactement les coupes de façon indépendante. Ils tiennent également compte de la continuité entre les coupes adjacentes par le biais d'un terme de régularisation.

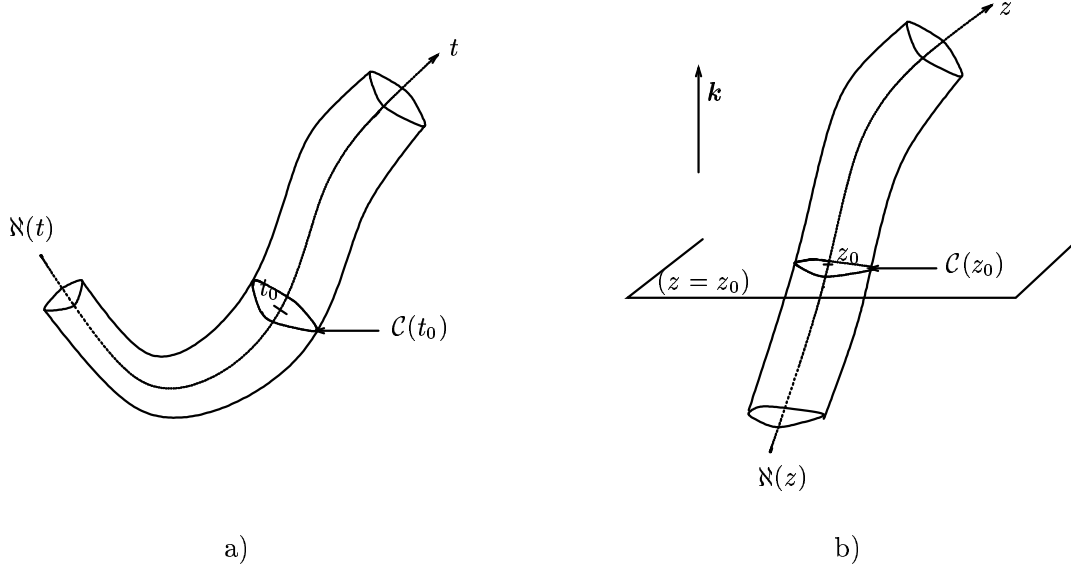


Figure III.2 – Modélisations 2,5D. a) Cylindre généralisé qui s'articule autour d'une courbe $N(t)$ de l'espace. Il est défini par les coupes transverses $\mathcal{C}(t) = \{\mathcal{C}(t, u), u \in (0, 2\pi)\}$, contours 2D fermés inclus dans les plans orthogonaux à N en chaque position $N(t)$, b) Cas où t décrit l'axe z : modélisation par tranches $\mathcal{C}(z)$ orthogonales à k .

Pour notre application, nous n'optons pas pour ces modèles pour deux raisons majeures, outre le fait qu'ils ne gèrent pas les parties inter-coupes. D'une part, l'objet que nous recherchons est supposé assez général, donc nous ne souhaitons pas l'orienter dans une directions particulière. D'autre part, la configuration des projections (voir la figure I.2 page 15) ne se prête pas à cette modélisation, car elles ne sont pas toutes orthogonales à un axe commun.

Modèles 3D. Nous envisageons donc les modèles paramétriques *réellement 3D*, que nous développons aux § III.4.2 et III.4.3. Nous choisissons les modèles harmoniques car ils sont étoilés et évitent de créer des formes qui s'auto-intersectent. De plus, ils génèrent des formes variées pour un nombre limité de paramètres. Dans un second temps, nous préconisons les contours splines pour leur caractère local. Concernant ces derniers, nous étudions les splines cubiques pour la simplicité de l'exposé, mais la reconstruction de splines d'ordre quelconque est aussi envisageable. Le choix de l'ordre, qui représente le degré de localité de la spline, sera finalement discuté au § III.5.

III.3.3 Estimation des paramètres

Dans les cas où le modèle ne dépend que d'un faible nombre de paramètres, nous réalisons l'estimation de $\mathbf{x}$ au sens du maximum de vraisemblance, éventuellement pénalisé. Par hypothèse, le bruit contenu dans les données est additif, blanc, gaussien et centré, cette estimation est donc identique à l'estimation au sens des moindres carrés (MC), qui s'écrit

$$\hat{\mathbf{x}}_{\text{MC}} = \arg \min_{\mathbf{x} \in \mathcal{X}} \{\mathcal{K}(\mathbf{x}) = \|\mathbf{d} - \mathcal{A}(\mathcal{C}_{\mathbf{x}})\|^2\}, \quad (\text{III.9})$$

où $\mathcal{X}$ est l'ensemble des contraintes sur $\mathbf{x}$, et $\mathcal{A}(\mathcal{C}_{\mathbf{x}})$ représente la projection du contour. Le contour reconstruit est donc $\mathcal{C}_{\hat{\mathbf{x}}_{\text{MC}}}$. Dans certains cas, il n'existe pas d'expression analytique de $\mathcal{A}(\mathcal{C}_{\mathbf{x}})$, ni

même d'algorithme de calcul exact de ce terme. Nous optons alors pour l'approximation de la forme $\mathcal{C}_{\mathbf{x}}$ par un polyèdre $\mathcal{P}_{\mathbf{x}}$ (un polygone dans le cas 2D), surface que nous savons projeter directement (voir les chapitres IV et V). Cette étape est résumée par la figure III.3⁴. Nous définissons finalement l'estimateur des moindres carrés approché (noté également $\hat{\mathbf{x}}_{\text{MC}}$) par

$$\hat{\mathbf{x}}_{\text{MC}} = \arg \min_{\mathbf{x} \in \mathcal{X}} \{ \tilde{\mathcal{K}}(\mathbf{x}) = \|\mathbf{d} - \mathcal{A}(\mathcal{P}_{\mathbf{x}})\|^2 \}. \quad (\text{III.10})$$

Pour les modèles à plus grand nombre de paramètres, l'estimation de $\mathbf{x}$ est en général mal posée car l'estimation au sens des MC est très sensible au bruit contenu dans les données. Pour obtenir une solution acceptable, il est indispensable d'introduire une information *a priori* sur le contour, *i.e.*, sur les paramètres $\mathbf{x}$. En l'absence d'hypothèse particulière sur la nature de $\mathcal{C}^*$, on choisit généralement une régularisation qui favorise la douceur du contour. En particulier, nous renvoyons le lecteur à (Fessler et Macovski, 1991; Lagrange, 1998) pour des modèles de douceur inter coupes liés à des formes 2,5D (pénalisation des formes dans lesquelles deux coupes adjacentes sont très différentes), et à (Hanson et al., 1994; Chalmond et al., 1994; Senasli et al., 2000) pour des douceurs relatives aux modèles par splines, développées par ailleurs aux chapitres IV et V pour les splines d'ordre 1. En utilisant le formalisme bayésien développé dans le § I.4, les paramètres $\mathbf{x}$ sont estimés au sens du MAP :

$$\hat{\mathbf{x}}_{\text{MAP}} = \arg \min_{\mathbf{x} \in \mathcal{X}} \{ \mathcal{J}(\mathbf{x}) = \mathcal{K}(\mathbf{x}) + \lambda \mathcal{R}(\mathbf{x}) \}, \quad (\text{III.11})$$

où $\mathcal{R}$ est la fonctionnelle de régularisation, et nous remplaçons $\mathcal{K}(\mathbf{x})$ par $\tilde{\mathcal{K}}(\mathbf{x})$ dans les cas où le calcul des projections de $\mathcal{C}_{\mathbf{x}}$ est approché. Cela revient donc à estimer le contour dont la projection est la plus proche des données, tout en satisfaisant les hypothèses *a priori*. Le paramètre λ règle le compromis entre les informations relatives aux données et à l'*a priori*. Un autre hyperparamètre est souvent associé à $\mathcal{R}$. Par exemple, nous utilisons aux chapitres IV et V un paramètre de seuil qui définit le degré de douceur des splines d'ordre 1. Ces paramètres sont fixés empiriquement.

Les critères $\mathcal{K}$ et $\mathcal{J}$ sont généralement non convexes. Néanmoins, pour des raisons de simplicité, nous prônons un algorithme déterministe de type gradient pour leur optimisation (voir § IV.3). Cette stratégie soulève des questions, notamment le choix de son initialisation si le critère n'est pas convexe. Ce point sera davantage discuté aux chapitres IV (§ IV.3), V (§ V.3) et VI.

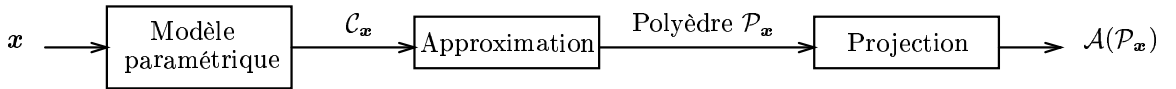


Figure III.3 – Mécanisme de projection approchée d'un contour paramétré.

⁴Lorsqu'une représentation continue $M(t, u)$ de $\mathcal{C}_{\mathbf{x}}$ est disponible, cette approximation consiste à fixer des valeurs particulières des paramètres (t, u) pour définir les sommets de $\mathcal{P}_{\mathbf{x}}$. Elle est précisée aux paragraphes suivants, notamment en III.4.2.

III.4 Étude des modèles retenus

Dans cette section, nous étudions de façon plus détaillée les modèles sélectionnés ci-dessus, en précisant le cas échéant nos contributions pour la procédure de reconstruction. De même que dans la partie III.2, nous présentons ces modèles en allant du plus simple au plus sophistiqué, en précisant à chaque fois l'apport du modèle décrit.

III.4.1 Ellipsoïdes et superquadriques

Le modèle ellipsoïdal pour les contours fermés 3D est très simple, car il possède 3 axes de symétrie et ne dépend que de 10 paramètres réels. Néanmoins, c'est un modèle intéressant, car c'est l'un des seuls dont les projections s'expriment analytiquement en fonction de $\mathbf{x}$. Dans un premier temps, nous le décrivons et proposons une procédure originale d'estimation de ses paramètres. Puis, nous étudions les super-ellipsoïdes, formes plus variées et non nécessairement convexes obtenues avec 2 paramètres réels supplémentaires.

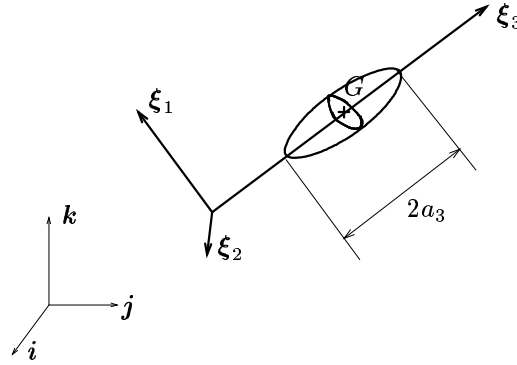


Figure III.4 – Paramétrisation d'un ellipsoïde.

Ellipsoïdes. Un ellipsoïde $\mathcal{C}_{\mathbf{x}}$ est une surface paramétrique convexe et étoilée autour d'un point. En notant G un tel point et $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3)$ l'orientation de l'ellipsoïde ($\boldsymbol{\xi}$ est une base orthonormée de $\mathbb{R}^3$ dont les vecteurs ξ_i sont les axes principaux d'inertie de $\mathcal{C}_{\mathbf{x}}$, ξ_3 correspondant au plus fort moment d'inertie), l'équation implicite de l'ellipsoïde dans la base $\boldsymbol{\xi}$ est

$$\sum_{i=1}^3 \frac{[\xi_i - \xi_i(G)]^2}{a_i^2} = 1, \quad (\text{III.12})$$

où a_1, a_2, a_3 sont les demi-longueurs de $\mathcal{C}_{\mathbf{x}}$, et $0 < a_1 \leq a_2 \leq a_3$ (voir la figure III.4). La paramétrisation s'exprime également de façon explicite, toujours dans la base $\boldsymbol{\xi}$:

$$\mathbf{GM}(\boldsymbol{\theta}) = [a_1 \cos \theta \sin \varphi, a_2 \sin \theta \sin \varphi, a_3 \cos \varphi]^t, \quad (\text{III.13})$$

où les angles sphériques $\boldsymbol{\theta} = (\theta, \varphi) \in (0, 2\pi) \times (0, \pi)$ sont définis sur la figure I.4 (page 18), et sont ici relatifs à $\boldsymbol{\xi}$. Les paramètres du contour sont finalement $\mathbf{x} = \{G, \boldsymbol{\xi}, a_1, a_2, a_3\}$, où $\boldsymbol{\xi}$ est défini par 4 paramètres angulaires (2 angles sphériques pour ξ_2 et ξ_3 , ξ_1 se déduisant de ces axes).

En pratique, nous n'estimons pas $\mathbf{x}$ par moindres carrés, mais optons pour la méthode des moments décrite en annexe B. L'intérêt de cette méthode est qu'elle évite l'estimation simultanée des paramètres G, ξ_i, a_i , qui sont inhomogènes. Son principe est de chercher, à partir des données, les moments géométriques d'ordre inférieur à 2 de $\mathcal{C}^*$:

$$\mu_{ijk} = \iiint_{\mathcal{C}^*} x^i y^j z^k dx dy dz, \quad i + j + k \leq 2. \quad (\text{III.14})$$

Or, pour un ellipsoïde $\mathcal{C}_{\mathbf{x}}$, le moment d'ordre 0 est son volume, les moments d'ordre 1 caractérisent la position de son centre d'inertie (de coordonnées $(\mu_{100}/\mu_{000}, \mu_{010}/\mu_{000}, \mu_{001}/\mu_{000})$), et les moments d'ordre 2 caractérisent les axes principaux d'inertie et les longueurs a_i . En effet, les vecteurs ξ_i sont les vecteurs propres de la matrice d'inertie

$$\begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}, \quad (\text{III.15})$$

et les coefficients a_i sont liés à ses valeurs propres. Les éléments de cette matrice sont les moments d'inertie centrés. Par exemple, $I_{xx} = \mu_{020} + \mu_{002} - \mu_{000} (y(G)^2 + z(G)^2)$ et $I_{xy} = \mu_{110} - \mu_{000} x(G) y(G)$.

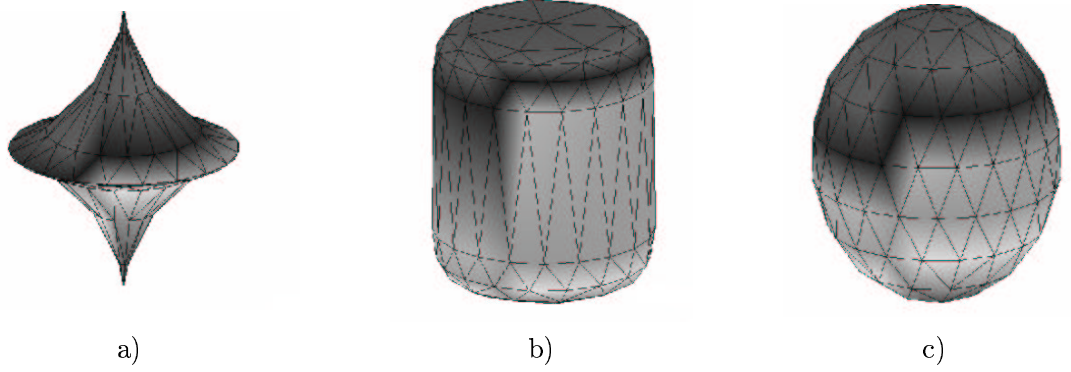


Figure III.5 – Paramétrisation des super-ellipsoïdes. (a,b) Super-ellipsoïdes orientés suivant l'axe (z) et générées avec $a_3 = 1, a_1 = a_2 = 1/2, \beta = 1$ et différentes valeurs de α ($\alpha = 3$ et 0.3 resp.). Les paramètres α et β correspondent à des contractions et dilatations suivant l'axe (z) et orthogonalement à l'axe (z) de l'ellipsoïde normalisé (c), pour lequel $\alpha = \beta = 1$.

Super-ellipsoïdes. Ces surfaces généralisent les ellipsoïdes, et sont définies par un très faible nombre de paramètres. En outre, elles sont étoilées par rapport à un point mais non nécessairement convexes, bien qu'elles possèdent des plans de symétrie. Avec les notations du paragraphe précédent, elles s'expriment implicitement par

$$\left\{ \left[\frac{\xi_1 - \xi_1(G)}{a_1} \right]^{2/\beta} + \left[\frac{\xi_2 - \xi_2(G)}{a_2} \right]^{2/\beta} \right\}^{\beta/\alpha} + \left[\frac{\xi_3 - \xi_3(G)}{a_3} \right]^{2/\alpha} = 1, \quad (\text{III.16})$$

et sous forme explicite, en coordonnées sphériques dans le repère ξ :

$$\mathbf{GM}(\theta) = [a_1 \cos^\beta \theta \sin^\alpha \varphi, a_2 \sin^\beta \theta \sin^\alpha \varphi, a_3 \cos^\alpha \varphi]^t, \quad (\text{III.17})$$

où α, β sont des paramètres positifs. Nous renvoyons le lecteur à (Solina et Bajcsy, 1990) pour plus de précisions. La forme des objets générés est illustrée en figure III.5 (a)-(b). En particulier, le cas $\alpha = \beta = 1$ correspond aux ellipsoïdes, et les variations de α (resp. β) déforment cet ellipsoïde longitudinalement, *i.e.*, suivant ξ_3 (resp. latéralement).

Pour reconstruire cette forme à partir des données, nous procédons au préalable à l'estimation de G et des axes d'inerties ξ par la méthode des moments. Le super-ellipsoïde est à présent paramétré

par $\mathbf{x} = [a_1, a_2, a_3, \alpha, \beta]^t$. L'estimation de $\mathbf{x}$ est réalisée au sens des moindres carrés contraints par (III.10), où l'ensemble des contraintes est

$$\mathcal{X} = \{(a_1, a_2, a_3) \in \mathbb{R}_+^3, 0 < a_1 < a_2 < a_3\} \times [0.2, 3]^2, \quad (\text{III.18})$$

la contrainte $(\alpha, \beta) \in [0.2, 3]^2$ permettant d'éviter la formation de surfaces dégénérées. À notre connaissance, il n'est pas possible de projeter ces contours de façon exacte. Néanmoins, cette famille est définie dans un système de coordonnées sphériques, et l'approximation de $\mathcal{C}_{\mathbf{x}}$ par un polyèdre $\mathcal{P}_{\mathbf{x}}$ dont les sommets sont définis tranche par tranche sur les plans ($\xi_3 = \text{constante}$) est simple et suffisamment précise. La minimisation de $\tilde{\mathcal{K}}(\mathbf{x})$ est effectuée par un algorithme de type gradient, avec une initialisation ellipsoïdale.

III.4.2 Contours harmoniques

Comme nous l'avons vu au § III.2.2.2, ces formes ont été principalement utilisées pour la segmentation et la vision d'objets 3D, mais pas pour la reconstruction tomographique. Dans cette section, nous synthétisons ces modèles en vue d'une application dans ce domaine. En particulier, nous motivons leur utilisation en remarquant qu'une sous-classe de ces modèles conduit à des formes *étoilées*, donc sans auto-intersection.

III.4.2.1 Contours 2D

Modélisation cartésienne. Nous décrivons le contour fermé 2D par un paramètre t , et exprimons la périodicité de la fonction $M(t) = (x(t), y(t))^t$. Sans perte de généralité, nous fixons sa période⁵ à 2π . La fonction $M(t)$ est continue et périodique, donc elle admet un développement en série de Fourier sous la forme

$$M(t) = \sum_{l=0}^{\infty} \mathbf{a}_l \cos(lt) + \sum_{l=1}^{\infty} \mathbf{b}_l \sin(lt), \quad \mathbf{a}_l, \mathbf{b}_l \in \mathbb{R}^2. \quad (\text{III.19})$$

Reconstruire $\mathcal{C}_{\mathbf{x}}$ revient donc à estimer les paramètres $\{\mathbf{a}_l, l \geq 0\}$ et $\{\mathbf{b}_l, l \geq 1\}$. En pratique, nous tronquons les deux sommes de (III.19) à l'ordre L de façon à estimer un ensemble fini de paramètres $\mathbf{x} = \{\mathbf{a}_l, \mathbf{b}_l, l \leq L\}$.

Ces modèles peuvent permettre une adaptation de la stratégie d'estimation de $\mathbf{x}$. En effet, nous pouvons envisager les estimations successives de $(\hat{\mathbf{a}}_1, \hat{\mathbf{b}}_1)$, puis de $(\hat{\mathbf{a}}_2, \hat{\mathbf{b}}_2)$ sachant $(\hat{\mathbf{a}}_1, \hat{\mathbf{b}}_1)$, et ainsi de suite. En d'autres termes, cela revient à considérer à l'étape $L' < L$ l'estimée basse fréquence

$$\sum_{l=0}^{L'} \hat{\mathbf{a}}_l \cos(lt) + \sum_{l=1}^{L'} \hat{\mathbf{b}}_l \sin(lt) \quad (\text{III.20})$$

comme l'approximation de l'estimée à l'étape $L' + 1$. Nous reviendrons sur ce choix au chapitre VI, où nous serons amenés à opter pour une estimation simultanée des paramètres $\mathbf{x}$ plutôt que pour ces estimations successives.

Dans l'ensemble, les modèles cartésiens présentent l'inconvénient de ne pas assurer la régularité de la courbe fermée au sens où elle ne s'intersecte pas. Nous montrons ci-dessous que certains modèles proches du modèle cartésien s'affranchissent de cette difficulté.

⁵Le paramètre t n'étant pas nécessairement l'abscisse curviligne s , nous ne sommes pas obligés de fixer la période au périmètre P de $\mathcal{C}_{\mathbf{x}}$. Pour un contour donné, nous définissons en fait t à partir de s par $t = 2\pi s/P$.

Modélisation polaire. Pour pallier ce défaut, nous décrivons les contours fermés dans un système de coordonnées polaires. En notant G un point arbitraire et $M(\theta)$ le point qui décrit la courbe fermée, ce modèle s'écrit

$$GM(\theta) = \rho(\theta) \mathbf{w}_\theta, \quad (\text{III.21})$$

où ρ est une fonction 2π -périodique. La troncature à l'ordre L du développement en série de Fourier de ρ s'écrit

$$\rho(\theta) = \sum_{l=0}^L a_l \cos(l\theta) + \sum_{l=1}^L b_l \sin(l\theta). \quad (\text{III.22})$$

Nous dérivons de ce modèle une condition suffisante très simple sur $\mathbf{x} = \{a_l, b_l, l \leq L\}$ pour que $\mathcal{C}_\mathbf{x}$ ne s'intersecte pas : $\forall \theta \in (0, 2\pi), \rho(\theta) > 0$. En imposant cette condition, nous ne décrivons plus l'ensemble des courbes planes fermées, mais uniquement celles qui sont étoilées par rapport à un point. La condition suffisante de régularité ne porte pas directement sur les coefficients a_l et b_l , mais nous la vérifions simplement en pratique, car nous échantillons la fonction $\rho(\theta)$ pour le calcul des projections du contour. Le polygone $\mathcal{P}_\mathbf{x}$ qui approxime $\mathcal{C}_\mathbf{x}$ est défini par les sommets $\{\mathcal{C}_\mathbf{x}(\theta_i = 2\pi i/n_\mathbf{x}), i = 1, \dots, n_\mathbf{x}\}$, et les contraintes de régularité (linéaires par rapport aux paramètres $\mathbf{x}$) s'expriment par $\mathcal{X} = \{\mathbf{x}, \forall i, \rho(\theta_i) > 0\}$.

Estimation de $\mathbf{x}$. Comme pour l'estimation de superquadriques, le choix du point G s'effectue en estimant la position du centre de gravité de l'objet à partir des données, par la méthode des moments. La décomposition (III.21) est effectuée dans le repère $\boldsymbol{\xi}$ formé par les deux axes principaux d'inertie estimés à partir des données (voir § III.4.1). Puis, les paramètres $\mathbf{x}$ sont estimés à partir des données au sens des moindres carrés contraints selon (III.10). L'initialisation de l'algorithme d'optimisation locale est circulaire, les cercles faisant effectivement partie de la classe des courbes paramétrées par (III.22) avec $L = 0$.

III.4.2.2 Surfaces 3D

Contours étoilés. Généralisant le modèle 2D, nous décrivons une surface compacte et étoilée par rapport à un point G . En nous plaçant dans le système de coordonnées sphériques relatif à un repère arbitraire $(G, \boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \boldsymbol{\xi}_3)$, l'expression de $\mathcal{C}_\mathbf{x}$ est :

$$GM(\boldsymbol{\theta}) = \rho(\boldsymbol{\theta}) \mathbf{w}_\boldsymbol{\theta}, \quad (\text{III.23})$$

où $\rho(\boldsymbol{\theta})$ est une fonction régulière et positive définie sur la boule unité de $\mathbb{R}^3$. Nous utilisons à présent le résultat suivant.

Proposition 3 (Matheny et Goldgof, 1995) *Toute fonction $\rho(\boldsymbol{\theta})$ d'énergie finie et différentiable sur la boule unité de $\mathbb{R}^3$ se décompose en une combinaison linéaire d'harmoniques sphériques⁶ :*

$$\rho(\boldsymbol{\theta}) = \sum_{l=0}^{\infty} \sum_{|k| \leq l} a_l^k \rho_l^k(\boldsymbol{\theta}), \quad (\text{III.24})$$

où les harmoniques sphériques $\rho_l^k(\boldsymbol{\theta})$ se déduisent des polynômes de Legendre⁷.

$$\rho_l^k(\boldsymbol{\theta}) = \begin{cases} P_l^k(\cos \theta) \cos(k\varphi) & \text{si } k \geq 0, \\ P_l^{|k|}(\cos \theta) \sin(|k|\varphi) & \text{sinon,} \end{cases} \quad (\text{III.25})$$

⁶De plus, les harmoniques sphériques ρ_l^k forment une famille orthogonale de l'espace des fonctions d'énergie finie sur la boule unité.

⁷Ces polynômes sont définis par $P_l(x) = 1/(2^l l!) \partial^l (x^2 - 1)^l / \partial x^l$.

où les fonctions de Legendre associées P_l^k sont définies à partir du polynôme de Legendre P_l :

$$P_l^k(x) = (-1)^k (1-x^2)^{k/2} \partial^k P_l(x) / \partial x^k.$$

En tronquant le développement (III.24) à l'ordre L , nous obtenons une paramétrisation par les $n = (L+1)^2$ coefficients $\mathbf{x} = \{a_l^k, |k| \leq l \leq L\}$.

Estimation de $\mathbf{x}$. De même qu'en 2D, le point G autour duquel la surface est étoilée est l'estimée du centre de gravité de l'objet (par la méthode des moments), et les paramètres $\mathbf{x}$ sont estimés au sens des moindres carrés contraints, où les contraintes sont du même type que dans le cas 2D. Pour le calcul de la projection de $\mathcal{C}_{\mathbf{x}}$, nous évaluons $\mathcal{A}(\mathcal{P}_{\mathbf{x}})$, projection d'une surface polyédrique approchant $\mathcal{C}_{\mathbf{x}}$. La discrétisation de $\mathcal{C}_{\mathbf{x}}$ est un peu plus technique que dans le cas 2D, au sens où un échantillonnage identique de $\theta \rightarrow M(\theta, \varphi)$ quel que soit l'angle φ n'est pas satisfaisant. En effet, si φ est égal à 0 ou π , la valeur de l'angle θ est sans importance car la position $M(\theta, \varphi)$ est indépendante de θ . En notant $\boldsymbol{\theta}_i = (\theta_i, \varphi_i)$, $i = 1, \dots, n_{\mathbf{x}}$ les angles sphériques associés aux sommets de $\mathcal{P}_{\mathbf{x}}$, la minimisation de $\tilde{\mathcal{K}}(\mathbf{x})$ est réalisée sous la contrainte (linéaire en $\mathbf{x}$) $\forall i, \rho(\boldsymbol{\theta}_i) > 0$.

III.4.3 Splines

En pratique, les modèles locaux sont plus adaptés que les précédents à la reconstruction de portions haute fréquence dans les contours, tout en générant des formes très variées. Cette remarque nous conduit au choix des splines, l'un des modèles locaux les plus classiques.

Sans perte de généralité, nous considérons les splines d'ordre 3, qui sont les plus couramment utilisées. Le choix de l'ordre ne modifie pas fondamentalement le raisonnement de cette étude, à l'exception du calcul analytique des projections (§ III.4.3.3). Nous reviendrons sur ce choix au § III.5. Soit donc une surface spline bicubique à n points de contrôle P_{ij} , dont nous estimons les coordonnées à partir des données. Les grandes différences avec les modèles développés aux paragraphes précédents sont les suivantes :

- Le modèle n'impose pas d'hypothèse particulière sur le contour, à l'exception de sa douceur locale.
- Le modèle est local par rapport aux paramètres. En d'autres termes, chaque point de contrôle n'influe que sur une portion du contour.
- Le calcul des projections est exact.

Nous reviendrons sur les deux derniers points dans les développements suivants. Nous traitons au départ les splines 2D, puis présentons leur extension au cas 3D. Enfin, nous développons l'estimation des paramètres du modèle et les difficultés algorithmiques associées.

III.4.3.1 Splines cubiques 2D

Définition. Étant donné $n+1$ abscisses $a = t_0 \leq t_1 \leq \dots \leq t_n = b$, une spline cubique de $\mathbb{R}^2$ est une courbe paramétrée $M(t) : (a, b) \rightarrow \mathbb{R}^2$ qui satisfait les conditions suivantes.

- (i) M est un polynôme de degré 3 sur chaque intervalle (t_i, t_{i+1}) .
- (ii) M est prolongée $\mathcal{C}^2$ en les points $t_1, \dots, t_{n-1}$.

Les splines cubiques forment un espace vectoriel de dimension $n+3$, dont une base est formée par les fonctions B-spline $B_i(t)$ de support (t_{i-2}, t_{i+2}) , $i = -1, 0, \dots, n+1$. La définition de ces fonctions nécessite de fixer arbitrairement 6 positions supplémentaires $t_{-3} \leq t_{-2} \leq t_{-1} (\leq t_0)$ puis $t_{n+3} \geq t_{n+2} \geq t_{n+1} (\geq t_n)$ ⁸. Finalement, les splines cubiques $M(t)$ se décomposent sous la forme

$$M(t) = \sum_{i=-1}^{n+1} P_i B_i(t), \quad t \in (a, b), \quad (\text{III.26})$$

⁸De plus, si les abscisses t_i sont équidistantes, les fonctions B_i se déduisent de B_0 par translation, i.e., $B_i(t) = B_0(t - ih)$, où h est le pas d'échantillonnage.

où les points $P_i \in \mathbb{R}^2$ sont appelés les *points de contrôle* de la spline. Compte-tenu du fait que les fonctions B_i ont des support bornés, chaque point de contrôle P_i n'a une influence que sur la partie de la courbe qui correspond au support de B_i , *i.e.*, l'intervalle (t_{i-2}, t_{i+2}) . D'où le caractère local du modèle (III.26)⁹.

Splines fermées. Pour définir une spline cubique *fermée*, nous ajoutons à (i) et (ii) la condition (iii) $M^{(k)}(a) = M^{(k)}(b)$, pour $k = 0, 1, 2$.

L'espace des splines cubiques fermées est donc de dimension n , et la condition (iii) se traduit sur les points de contrôle P_i de l'expression (III.26) : $P_i = P_{n+i}$ pour $i = -1, 0, 1$. Le modèle des splines cubiques fermées devient

$$M(t) = \sum_{i=1}^n P_i B_i(t), \quad t \in \mathbb{R}, \quad (\text{III.27})$$

en prolongeant par périodicité les fonctions $B_i(t)$ sur $\mathbb{R}$ par $\forall k \in \mathbb{Z}, B_i(t + k(b-a)) = B_i(t)$.

III.4.3.2 Surfaces splines

L'extension du modèle précédent à une fonction bidimensionnelle $M(t, u)$ à valeurs dans $\mathbb{R}^3$ ne pose pas de difficultés particulières (Chalmond, 2000, chapitre 3). Il s'agit cette fois de définir une grille $\{t_0, \dots, t_q\} \times \{u_0, \dots, u_q\}$. Par simplicité, nous supposons que cette grille possède $(q+1)^2$ positions échantillonnées de façon uniforme sur (a, b) , où $a = t_0 = u_0$, $b = t_q = u_q$ (donc $\forall i, t_i = u_i$). On démontre que l'espace des surfaces splines bicubiques fermées est de dimension $n = q^2$, et que sa base est formée par le produit des bases $\{B_i, i = 1, \dots, q\}$ relatives aux positions t_i et u_j . En d'autres termes, toute surface spline cubique s'exprime par

$$M(t, u) = \sum_{i=1}^q \sum_{j=1}^q P_{ij} B_i(t) B_j(u), \quad (t, u) \in \mathbb{R}^2, \quad (\text{III.28})$$

où les positions $P_{ij} \in \mathbb{R}^3$ sont les points de contrôle de la spline, et les fonctions B_i sont étendues par périodicité par $\forall k \in \mathbb{Z}, B_i(t + k(b-a)) = B_i(t)$. De même que dans le cas 2D, chaque point P_{ij} contrôle uniquement une portion locale de la surface, correspondant à $(t, u) \in (t_{i-2}, t_{i+2}) \times (u_{j-2}, u_{j+2})$.

III.4.3.3 Reconstruction de splines fermées

Dans les cas 2D et 3D, le vecteur des paramètres $\mathbf{x}$ est formé de l'ensemble des coordonnées des n points de contrôle P_i ou P_{ij} de la spline. Le caractère local des paramètres est intéressant car il permet de modéliser des parties « haute fréquence » dans un contour globalement basse fréquence. L'estimation de $\mathbf{x}$ est réalisée au sens des MC ou du MAP. Nous soulignerons en particulier aux chapitres IV, V et VI l'importance de la régularisation pour les modèles splines d'ordre 1, lorsque n est suffisamment grand.

À présent, nous abordons un point que les modèles harmoniques ne permettent pas de résoudre : le calcul exact des projections. Nous avons vu au § III.1 qu'il est directement lié aux intersections des rayons de projection $(S, \mathbf{w})$ avec $\mathcal{C}_{\mathbf{x}}$. En 2D, cela revient à rechercher les paramètres t tels que $P(t) = \langle \mathbf{S}M(t), \mathbf{w}^\perp \rangle = 0$, où $M(t)$ est décrit par (III.27), et $\mathbf{w}^\perp$ est le vecteur unitaire de $\mathbb{R}^2$ directement orthogonal à $\mathbf{w}$. Comme la restriction de P à chaque intervalle (t_i, t_{i+1}) est un polynôme de degré inférieur à 3 (P est en fait une fonction spline cubique 1D), la recherche des racines de P sur chaque intervalle s'effectue de façon analytique, d'où le calcul exact des projections de la spline.

⁹En fait, dans un cadre général, le degré de localité dépend de l'ordre p de la spline, *i.e.*, le degré des polynômes B_i . Plus p est élevé, plus le support des B-splines est étendu. Par exemple, si les positions t_i sont équidistantes, la longueur de ce support est $(p+1)h$ en notant h le pas d'échantillonnage.

Dans le cas 3D, un raisonnement analogue mène à la résolution d'un système de deux équations à deux inconnues, en faisant intervenir la base orthonormée $(\mathbf{u}, \mathbf{v}, \mathbf{w})$ définie sur la figure I.4 (page 18).

$$\begin{cases} \langle \mathbf{SM}(t, u), \mathbf{u} \rangle = \sum_{i,j} \langle \mathbf{SP}_{ij}, \mathbf{u} \rangle B_i(t) B_j(u) = 0, \\ \langle \mathbf{SM}(t, u), \mathbf{v} \rangle = \sum_{i,j} \langle \mathbf{SP}_{ij}, \mathbf{v} \rangle B_i(t) B_j(u) = 0. \end{cases} \quad (\text{III.29})$$

Cette fois, nous ne pouvons pas obtenir simplement une expression analytique des solutions (t, u) , bien qu'à u fixé, la première équation de (III.29) mène à une expression analytique de $t = \Phi(u)$. En pratique, la résolution de (III.29) s'effectue numériquement.

III.5 Conclusions

Dans ce chapitre, nous avons étudié divers modèles paramétriques de courbes ou surfaces fermées pour la reconstruction tomographique. Indépendamment des modèles, les difficultés techniques que nous rencontrons sont les suivantes : le calcul exact des projections et la gestion des auto-intersections du contour. Pour le premier point, nous proposons des approximations splines si les modèles sont « non projetables » analytiquement, les splines d'ordre faible étant elles projetables. En particulier, le calcul des projections d'un polyèdre à facettes triangulaires, détaillé au chapitre V, est très simple car il ne fait intervenir que des intersections rayons-facettes.

Parmi les modèles paramétriques, les modèles globaux sont les plus simples, mais peu appropriés à la reconstruction précise de contours complexes, car ils ne retiennent que les composantes basse fréquence des contours. En particulier, si l'on considère des déformations locales d'un contour paramétré de référence, l'adaptation du modèle nécessite le réajustement de la *totalité* de ses paramètres. En dépit de ces limites, nous retenons les modèles globaux, et notamment le modèle harmonique, pour fournir une reconstruction « grossière » du contour. Nous utiliserons aux chapitres IV et V (§ V.3) cette reconstruction comme l'initialisation d'une méthode locale plus élaborée. Le choix du modèle harmonique est motivé par la diversité des formes qu'il engendre (formes non convexes, bien qu'étoilées), contrairement aux ellipsoïdes et super-ellipsoïdes.

Les modèles locaux offrent un meilleur compromis entre la diversité des formes générées et le nombre de paramètres. En revanche, ces modèles sont « haute fréquence », donc nécessitent un plus grand nombre de paramètres, et mènent à des méthodes d'estimation complexes (introduction d'information *a priori*). Parmi ces modèles, nous optons pour les splines fermées, en estimant les positions de leurs points de contrôle directement à partir des données. Le degré de localité dépend de l'ordre de la spline, c'est-à-dire du degré des fonctions B-splines B_i qui interviennent dans les décompositions (III.27) et (III.28). Pour des points de contrôle P_{ij} fixés, cet ordre est lié à la portion de la spline contrôlée par chaque point P_{ij} (plus l'ordre est important, plus les aires de ces portions sont importantes). Nous choisissons finalement une spline d'ordre 1, c'est-à-dire un *polygone* en 2D ou un *polyèdre* en 3D. Ce choix est guidé par le fait qu'une spline d'ordre faible impose peu de contraintes de douceur locale, et modélise plus simplement les composantes locales haute fréquence du contour. Mais ce choix est surtout guidé par des considérations algorithmiques, en particulier la simplicité de l'évaluation et de la remise à jour des projections. Ce dernier modèle est plus amplement détaillé aux chapitres IV et V.

CHAPITRE IV

APPROCHE POLYGONALE

- IV.1 Introduction
- IV.2 Modèle polygonal
- IV.3 Optimisation du critère
- IV.4 Étude des projections
- IV.5 Conclusions
- IV.A Caractère non convexe de $\mathcal{K}$

IV.1 Introduction

Dans le chapitre III, nous avons étudié divers modèles géométriques déformables pour la reconstruction de contours fermés. Nous avons finalement opté pour les modèles *spline* d'ordre 1 pour plusieurs raisons : la grande diversité des formes qu'ils engendrent, le caractère local des paramètres, et la facilité de l'évaluation de leur projection en un détecteur ponctuel. Dans les chapitres IV et V, nous nous focalisons sur ces représentations polygonale et polyédrique pour les problèmes respectifs de la reconstruction de formes compactes à partir de données radiographiques 2D et 3D. En notant $\mathcal{P}$ une telle forme, nous rappelons que la reconstruction se ramène à l'estimation des paramètres de $\mathcal{P}$, *i.e.*, les coordonnées de ses sommets, à partir des données radiographiques $\mathbf{d}$.

Dans la littérature, ces formes ont été largement utilisées en segmentation d'image, avec l'avènement de la méthode des éléments finis (Davies, 1980), mais plus rarement pour des problèmes de reconstruction, où les données (les valeurs des pixels des images projections) ne sont pas reliées directement aux sommets de $\mathcal{P}$. Dans ce domaine, les contributions récentes qui visent à estimer directement les sommets de $\mathcal{P}$ sont notamment (Milanfar et al., 1994; Hanson et Cunningham, 1995; Hanson et al., 1996, 1997; Mohammad-Djafari, 1997) en 2D et (Battle et al., 1998; Mohammad-Djafari et Soussen, 1999; Soussen et Mohammad-Djafari, 1999a) en 3D. Bien que nous développons les splines d'ordre 1 à des fins tridimensionnelles, nous préférons traiter le cas 2D, plus simple, avant de l'étendre au chapitre V. Les difficultés du passage au cas 3D sont de plusieurs ordres. Parfois, ce ne sont que des adaptations techniques. Sur d'autres points, l'extension impose des limitations d'ordre théorique.

Ce chapitre s'organise comme suit. Tout d'abord, nous définissons le modèle polygonal et développons différentes méthodes d'estimation (§ IV.2). En particulier, nous commençons par étudier la méthode dite « des moments » (Milanfar et al., 1994, 1995, 1996). Nous exprimons ses limites, pour motiver une procédure d'estimation plus sophistiquée. Il s'agit d'une approche régularisée qui réalise une estimation directe des sommets de $\mathcal{P}$ à partir des projections (Hanson et Cunningham, 1995; Mohammad-Djafari, 1997; Battle et al., 1998; Mohammad-Djafari et Soussen, 1999). Par la suite, nous analysons les problèmes d'optimisation qui en découlent, et optons pour un algorithme de descente par blocs, où les blocs représentent les sommets (§ IV.3). Puis, la section IV.4 est consacrée aux problèmes de mise en œuvre associés. Ces problèmes concernent principalement les manipulations des projections (leur calcul, leur remise à jour, et l'évaluation de leur gradient par rapport aux sommets) et la gestion du caractère « non intersecté » des polygones reconstruits.

IV.2 Modèle polygonal

Le polygone $\mathcal{P}$ est défini par la séquence de ses sommets v_i , ordonnés dans le sens trigonométrique :

$$\mathcal{P} \equiv \mathbf{v} = [v_1, \dots, v_n]^t, \quad v_i \in \mathbb{C}. \quad (\text{IV.1})$$

Par circularité, nous dénotons $v_0 \equiv v_n$ et $v_{n+1} \equiv v_1$, puis nous définissons l'ensemble des polygones *acceptables*, sous-ensemble de $\mathbb{C}^n$ noté $\mathbb{A}_n$.

$$\mathbb{A}_n = \left\{ \mathbf{v} \in \mathbb{C}^n, \forall (i, j) \in \{1, \dots, n\}, |i - j| > 1, (v_i, v_{i+1}) \cap (v_j, v_{j+1}) = \emptyset \right\}. \quad (\text{IV.2})$$

Cet ensemble représente les polygones qui ne s'intersectent pas (cf. figure IV.1). Nous rappelons que la fonction de densité de la scène est définie par

$$f(x, y) = \begin{cases} 1 & \text{si } M(x, y) \in \mathcal{P}, \\ 0 & \text{sinon.} \end{cases} \quad (\text{IV.3})$$

Le problème de la reconstruction de la scène est donc équivalent à l'estimation de $\hat{\mathbf{v}} \in \mathbb{A}_n$ à partir des données $\mathbf{d}$.

Dans la suite, nous étudions la méthode des moments, qui consiste à estimer à partir des données les moments géométriques de l'objet jusqu'à un certain ordre, puis dans un deuxième temps à estimer $\hat{\mathbf{v}}$ à partir des moments. Nous exprimons ses limites, pour proposer une approche régularisée au § IV.2.2.

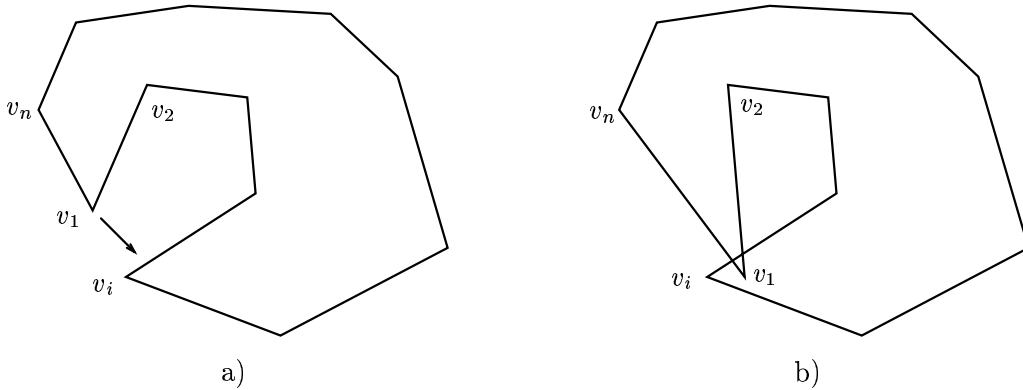


Figure IV.1 – Exemple d'auto-intersection consécutive au déplacement d'un sommet de $\mathbf{v}$.

IV.2.1 Méthode des moments

Cette méthode, développée par Milanfar et al. (1994, 1995, 1996), fait intervenir les moments géométriques de $\mathcal{P}$ définis par

$$\mu_{ij} = \iint_{\mathbb{R}^2} x^i y^j f(x, y) dx dy = \iint_{\mathcal{P}} x^i y^j dx dy. \quad (\text{IV.4})$$

Elle consiste, pour un ordre p fixé, à :

1. Estimer les moments $\boldsymbol{\mu}^k = \{\mu_{ij}, i + j = k\}$, $k \leq p$, à partir des données radiographiques $\mathbf{d}$.
2. Reconstruire $\hat{\mathcal{P}} = \hat{\mathbf{v}}$ à partir des estimées $\hat{\boldsymbol{\mu}}^0, \dots, \hat{\boldsymbol{\mu}}^p$ (et éventuellement des données).

Le premier sous-problème a été étudié dans (Milanfar et al., 1995), qui établissent une relation analytique et linéaire entre le vecteur $\boldsymbol{\mu}^k$ et les moments 1D d'ordre k des projections $p_\theta(t)$ de $\mathcal{P}$. Ils étudient l'inversibilité de cette relation à partir de $k + 1$ projections distinctes $\theta_1, \dots, \theta_{k+1}$.

L'estimation de $\boldsymbol{\mu}^0, \boldsymbol{\mu}^1, \dots, \boldsymbol{\mu}^p$ est finalement réalisée au sens des moindres carrés à partir d'au moins $p+1$ données. Pour plus de précisions, nous renvoyons le lecteur au § B.2 de (Soussen et Mohammad-Djafari, 1999c), que nous intégrons dans l'annexe B.

Le second sous-problème, *i.e.*, l'estimation de $\hat{\mathbf{v}}$ à partir de ses moments $\hat{\boldsymbol{\mu}}^0, \dots, \hat{\boldsymbol{\mu}}^p$, a donné lieu à plusieurs développements. Pour des applications en tomographie de diffraction, Buvaneswari et Naidu (1998) ont étudié un problème inverse similaire, où chaque donnée observée mesure le champ diffracté sur un cercle $\mathcal{C}$ du domaine fréquentiel. Elle correspond à la transformée de Fourier 2D de f sur ce cercle :

$$\forall (\nu_x, \nu_y) \in \mathcal{C}, \quad \mu(\nu_x, \nu_y) = \iint_{\mathcal{P}} \exp[-2i\pi(\nu_x x + \nu_y y)] dx dy. \quad (\text{IV.5})$$

Les auteurs reconstruisent un polygone $\mathcal{P}$ *convexe*. Un tel choix est guidé par le faible nombre de données (3 mesures), et permet en outre la décomposition de $\mathcal{P}$ en une réunion de triangles étoilés par rapport à un point intérieur de $\mathcal{P}$, et donc un calcul analytique de (IV.5) en fonction de $\mathbf{v}$, puis l'inversion d'une telle relation. Milanfar et al. (1995) traitent aussi le sous-problème 2 (après avoir résolu le sous-problème 1), mais dans un cadre plus général où le polygone est *quelconque*. Ils établissent l'évaluation exacte des sommets $v_1, \dots, v_n$ de $\mathcal{P}$ à partir d'au moins $2n-2$ moments : $\boldsymbol{\mu}^0, \dots, \boldsymbol{\mu}^{2n-3}$. Néanmoins, seules les positions des sommets de $\mathcal{P}$ sont déterminées, et leur ordre doit être recherché par une procédure complémentaire que nous ne détaillons pas.

La conclusion des travaux précédents est que la reconstruction de $\mathcal{P}$ est réalisable par la méthode des moments à partir d'au moins $2n-2$ projections. Cependant, cette méthode se limite en pratique aux polygones à peu de sommets¹ dans les situations où les données sont en nombre limité. De plus, le sous-problème 1 est mal posé car les estimations $\hat{\boldsymbol{\mu}}^k$ sont instables vis à vis des données, d'autant plus que k est grand. En d'autres termes, les estimées des moments géométriques d'ordre supérieur apportent peu d'information sur $\mathcal{P}$, sauf si le rapport signal à bruit des données est fort. Finalement, Milanfar et al. procèdent en deux temps. Tout d'abord, ils reconstruisent un polygone $\mathcal{P}^{(0)}$ *affinement régulier*, donc convexe, à partir de $\boldsymbol{\mu}^0 = \{\mu_{00}\}$, $\boldsymbol{\mu}^1$, $\boldsymbol{\mu}^2$. Puis, ils considèrent $\mathcal{P}^{(0)}$ comme l'initialisation d'une méthode de reconstruction plus raffinée (approche MDL décrite au § IV.2.2.1).

Milanfar et al. (1996) ont aussi étudié le sous-problème 2 dans un contexte variationnel. Ils considèrent la reconstruction d'une carte de densité $f(x, y)$ non binaire et définie sur un domaine continu, à partir des moments $\boldsymbol{\mu} = \{\boldsymbol{\mu}^k, k = 1, \dots, p\}$. Pour cela, ils estiment $f(x, y)$ au sens du MAP :

$$\hat{f}_{\text{MAP}}(x, y) = \arg \min_f \{ -\log p(\boldsymbol{\mu} | \boldsymbol{\mu}(f)) - \log p(f) \}. \quad (\text{IV.6})$$

Le terme d'attache aux données est $-\log p(\boldsymbol{\mu} | \boldsymbol{\mu}(f)) \propto (\boldsymbol{\mu} - \boldsymbol{\mu}(f))^t \Sigma_p (\boldsymbol{\mu} - \boldsymbol{\mu}(f))$, où Σ_p représente l'inverse de la matrice de covariance de l'estimée $\boldsymbol{\mu}$, et la loi *a priori* $p(f)$ est de nature entropique.

Pour notre part, nous optons pour l'approche la plus simple, *i.e.*, la reconstruction de polygones convexes (Milanfar et al., 1995). Elle s'effectue à partir de $\mu_{00}, \boldsymbol{\mu}^1, \boldsymbol{\mu}^2$, et s'appuie sur un polygone régulier de référence $\mathbf{v}_{\text{REF}} = \{v_1^{\text{REF}}, \dots, v_n^{\text{REF}}\}$ d'aire unité, et une transformation affine $\mathbf{L}$ qui associe $\mathbf{v}$ à $\mathbf{v}_{\text{REF}}$:

$$(x_i, y_i)^t = \mathbf{g} + \mathbf{L} (x_i^{\text{REF}}, y_i^{\text{REF}})^t, \quad i = 1, \dots, n, \quad (\text{IV.7})$$

où $\mathbf{g} \in \mathbb{R}^2$ est le vecteur translation et la matrice $\mathbf{L}$ de taille 2×2 réalise une transformation affine du polygone régulier. Les éléments de $\mathbf{L}$ sont déterminés de façon unique à partir des moments $\mu_{00}, \boldsymbol{\mu}^1, \boldsymbol{\mu}^2$. En effet, μ_{00} et $\boldsymbol{\mu}^1$ définissent le centre de gravité de $\mathcal{P}$, *i.e.*, $\mathbf{g} = (\mu_{10}/\mu_{00}, \mu_{01}/\mu_{00})^t$, et les moments d'ordre 2 permettent de remplir la matrice $\mathbf{L}$.

¹Milanfar et al. reconstruisent des polygones à moins de 5 sommets.

IV.2.2 Approche proposée

Dans la suite, nous développons une approche dont le but est de reconstruire des polygones à plus grand nombre de sommets. Nous dénotons les projections de $\mathcal{P} \equiv \mathbf{v}$ (projections calculées en les pixels détecteurs considérés comme ponctuels, voir le chapitre I) par :

$$\mathbf{p} = \mathcal{A}(\mathbf{v}) = \mathcal{A}(v_1, \dots, v_n), \quad (\text{IV.8})$$

où l'opérateur $\mathcal{A}$ est défini sur $\mathbb{A}_n^2$. Le terme d'adéquation aux données est donc

$$\mathcal{K}(\mathbf{v}) = \|\mathbf{d} - \mathcal{A}(\mathbf{v})\|^2. \quad (\text{IV.9})$$

Malgré le caractère affine par morceaux, donc très simple des projections $t \rightarrow p_\theta(t)$ d'un polygone, il est important de noter que, contrairement au modèle par pixels,

- $\mathbf{v} \rightarrow \mathcal{A}(\mathbf{v})$ n'est pas un opérateur affine ;
- $\mathcal{A}(\mathbf{v}) = \mathcal{A}(x_1, \dots, x_n, y_1, \dots, y_n)$ n'est pas différentiable partout sur $\mathbb{A}_n$; et par suite,
- $\mathcal{K}(\mathbf{v})$ est en général non convexe et multimodal sur $\mathbb{A}_n$.

La première et la troisième assertions sont illustrées sur un exemple simple dans l'annexe IV.A, la seconde est détaillée en annexe C.

IV.2.2.1 Estimation de $\hat{\mathbf{v}}$ par moindres carrés

L'estimation au sens des moindres carrés (MC) est équivalente à l'estimation au sens du maximum de vraisemblance car nous supposons que les erreurs du modèle direct sont blanches, centrées et gaussiennes. L'estimateur $\hat{\mathbf{v}}_{\text{MC}}$ est défini par :

$$\hat{\mathbf{v}}_{\text{MC}} = \arg \min_{\mathbf{v} \in \mathbb{A}_n} \{-\log p(\mathbf{d} | \mathbf{v})\} = \arg \min_{\mathbf{v} \in \mathbb{A}_n} \mathcal{K}(\mathbf{v}). \quad (\text{IV.10})$$

C'est l'estimateur utilisé par Milanfar et al. (1994), dans le cadre d'un modèle à très faible nombre de sommets (inférieur à 10), à la différence du nôtre. Dans un second temps, ces auteurs étudient le problème de la reconstruction d'un polygone à nombre variable de sommets. À cet effet, ils ont recours à une approche de type MDL (*Minimum Description Length*) qui pénalise les configurations à grand nombre de sommets. Ils sont donc amenés à rechercher conjointement

$$(\hat{\mathbf{v}}_{\text{MDL}}, \hat{n}) = \arg \min_{\mathbf{v}, n} \{-\log p(\mathbf{d} | \mathbf{v}) + n \log(M)/2\}. \quad (\text{IV.11})$$

Cette minimisation est réalisée en recherchant pour tout n , l'estimateur au sens des moindres carrés $\hat{\mathbf{v}}_{\text{MC}}(n)$, puis en sélectionnant $\hat{n}$ à partir de ces configurations.

Revenons à présent à l'évaluation de $\hat{\mathbf{v}}_{\text{MC}}$. Cet estimateur minimise un critère non convexe et multimodal en général. Nous renvoyons le lecteur au § IV.3 pour le choix de l'algorithme d'optimisation. En particulier, nous développerons dans cette section la question de l'obtention de solutions acceptables au sens de $\mathbb{A}_n$. Enfin, nous verrons au chapitre VI que la qualité des résultats obtenus par moindre carrés est liée au nombre de sommets de $\mathbf{v}$, et qu'à grand nombre de sommets, l'estimateur $\hat{\mathbf{v}}_{\text{MC}}$ possède des irrégularités spatiales. Pour pallier ce défaut, nous considérons une approche régularisée, où nous apportons des informations *a priori* structurelles de douceur spatiale du polygone.

²Par abus de notation, nous notons également $\mathcal{A}(\mathbf{v}) = \mathcal{A}(x_1, \dots, x_n, y_1, \dots, y_n)$ l'opérateur défini sur $\mathbb{R}^{2n}$ et dont l'entrée correspond aux coordonnées des sommets.

IV.2.2.2 Estimation de $\hat{v}$ au sens du MAP

Principe. Dans le cadre de nos hypothèses, l'estimation au sens du MAP ne diffère de l'estimation par moindres carrés que par l'apport d'une information *a priori* sur la *douceur locale* du contour de l'objet. Il s'agit d'un choix classique en contours actifs pour la segmentation d'images (Blake et Zisserman, 1987; Kass et al., 1988), déjà évoqué au chapitre III. Dans ce domaine, on considère des courbes paramétrées $M(t) = u(t)$, $t \in (0, P)$ et on définit des pénalisations qui portent sur les dérivées successives du contour u , fonction au moins $\mathcal{C}^2$. En considérant les dérivabilités d'ordre 1 et 2, ces pénalisations sont du type :

$$\mathcal{R}(u) = \lambda_1 \int_0^P \|u'(t)\|^2 dt + \lambda_2 \int_0^P \|u''(t)\|^2 dt, \quad (\text{IV.12})$$

où le premier facteur est un terme d'élasticité ou de continuité locale, et le second est un terme de rigidité, qui porte sur la courbure locale de $M(t)$. C'est un terme qui pénalise en particulier les « coins » dans le contour. Avant d'étendre le schéma (IV.12) aux polygones, nous réalisons un bref rappel sur la paramétrisation normale d'un contour.

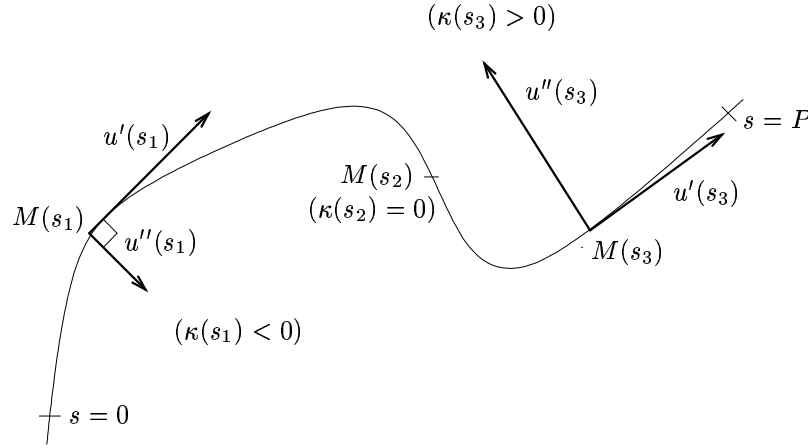


Figure IV.2 – Paramétrisation normale d'une courbe régulière (de classe $\mathcal{C}^2$) $M(s) = u(s)$. Illustration de la position des tangentes et normales à la courbe, et du signe des courbures $\kappa(s)$.

Paramétrisation normale. La paramétrisation normale d'un contour $\mathcal{C}$ correspond au cas où t est l'abscisse curviligne de $\mathcal{C}$, que nous notons s dorénavant. Cela signifie que s est la mesure de la longueur de l'arc $(M(0), M(s))$ (et P représente alors le périmètre de $\mathcal{C}$). Une telle paramétrisation induit les propriétés suivantes :

$$\begin{aligned} \forall s \in (0, P), \quad \|u'(s)\| &= 1, \\ \forall s \in (0, P), \quad \langle u'(s), u''(s) \rangle &= 0. \end{aligned} \quad (\text{IV.13})$$

En d'autres termes, $u'(s)$ est la tangente unitaire à $\mathcal{C}$ en $M(s)$, et $u''(s)$ est une normale associée (en général non unitaire), orientée vers le centre de courbure de $\mathcal{C}$ en $M(s)$ (voir la figure IV.2). De plus, la courbure de $\mathcal{C}$ en s est définie par

$$\kappa(s) = \varepsilon(s) \|u''(s)\|, \quad \text{où } \varepsilon(s) = \pm 1. \quad (\text{IV.14})$$

Plus précisément, $\varepsilon(s) = 1$ si et seulement si le système $(u'(s), u''(s))$ est direct.

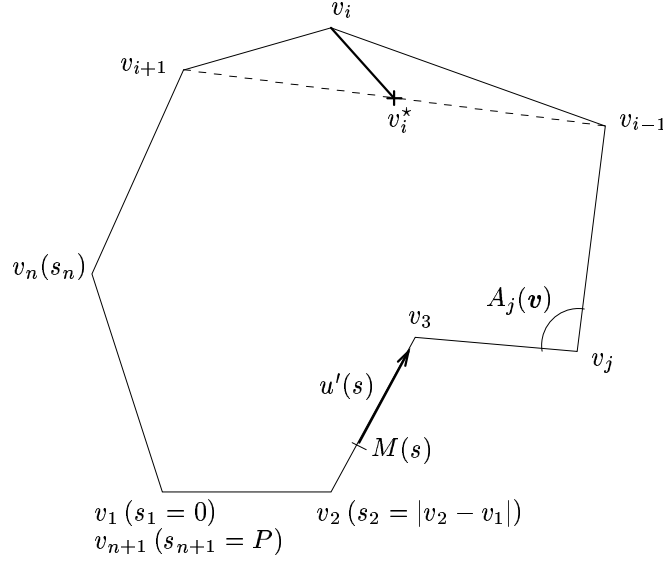


Figure IV.3 – Paramétrisation normale d'un polygone, et notions relatives à sa courbure.

Remarque. En revenant sur la première propriété énoncée en (IV.13), nous déduisons qu'une paramétrisation normale mène à

$$\int_0^P \|u'(s)\|^2 ds = P. \quad (\text{IV.15})$$

En d'autres termes, le terme d'élasticité de (IV.12) correspond au périmètre du contour fermé. Quant au terme du second ordre, il correspond à l'intégrale sur $\mathcal{C}$ de sa courbure non signée.

Paramétrisation d'un polygone. Pour un polygone de $\mathbb{A}_n$, la fonction u est injective (le polygone ne s'intersecte pas), et vérifie $u(0) = u(P)$ (c'est un contour fermé). Par contre, u n'est pas $\mathcal{C}^2$, mais simplement affine par morceaux. Par conséquent, nous ne pouvons pas écrire la seconde intégrale de la pénalisation (IV.12). En définissant les abscisses curvilignes des sommets de $\mathbf{v}$ par $s_i = \sum_{k=1}^{i-1} |v_{k+1} - v_k|$ (et $s_1 = 0$), puis la fonction $u(s)$ sur $(0, s_{n+1})$ par

$$\forall i \in \{1, \dots, n\}, \quad u(s) = v_i + (s - s_i) \frac{(v_{i+1} - v_i)}{s_{i+1} - s_i} \quad \text{pour } s \in (s_i, s_{i+1}), \quad (\text{IV.16})$$

la tangente au contour est $u'(s) = (v_{i+1} - v_i)/(s_{i+1} - s_i)$ pour $s \in (s_i, s_{i+1})$ (voir la figure IV.3). Compte tenu des remarques du paragraphe précédent, le terme d'élasticité de (IV.12) porte sur le périmètre $P = s_{n+1}$ du polygone. Nous préférons ne pas faire intervenir ce terme, car il a pour effet de favoriser les contours à longueur minimale (les minima d'une telle énergie étant les polygones réduits à un point³). Quant au terme du second ordre, il pénalise les fortes courbures du polygone. Pour le définir, nous envisageons une adaptation discrète de la seconde intégrale de (IV.12) par :

$$\mathcal{R}(\mathbf{v}) = \sum_{i=1}^n \phi(\kappa_i(\mathbf{v})), \quad (\text{IV.17})$$

où les termes $\kappa_i(\mathbf{v})$ sont les courbures approchées non signées en les sommets v_i (ils ne tiennent donc pas compte du caractère intérieur ou extérieur de v_i), et ϕ est une fonction paire et croissante sur $\mathbb{R}_+$. Nous revenons sur le choix de ϕ au dernier paragraphe de la présente section. Avant d'explicitier les termes $\kappa_i(\mathbf{v})$, nous rappelons que, dans l'interprétation bayésienne de la régularisation, la loi *a priori*

³Mohammad-Djafari (1997) propose une énergie de régularisation de la forme $\phi(\sum |v_{i+1} - v_i| - 2\pi R_0)$ lorsque l'on dispose d'une information *a priori* sur la taille du polygone, ce qui n'est pas notre cas.

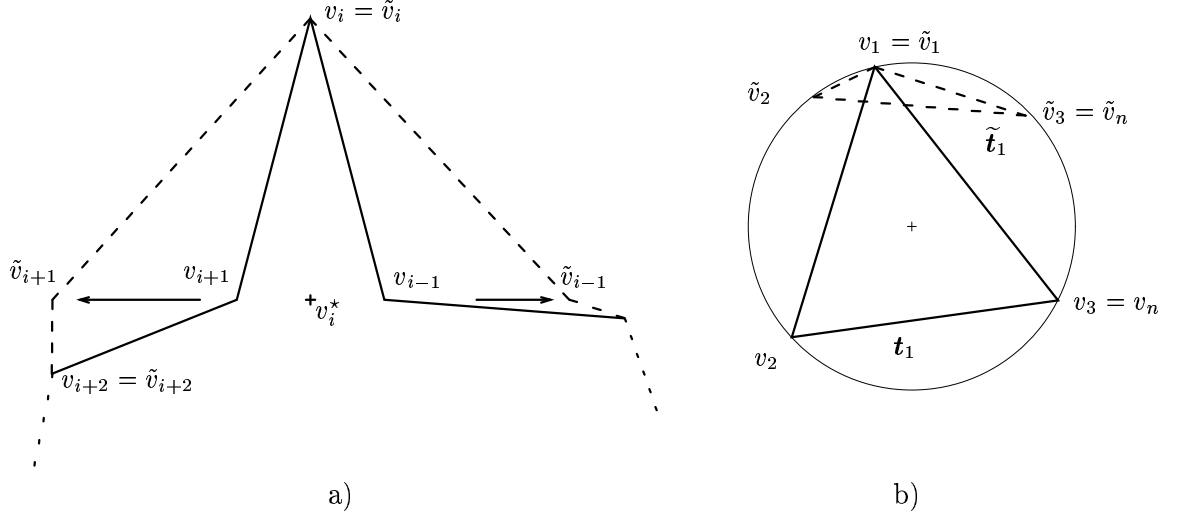


Figure IV.4 – Limite de la régularisation par la distance entre les sommets de $\mathbf{v}$ et les barycentres de leurs voisins (a), et par le rayon des cercles circonscrits aux triangles $\mathbf{t}_i$ (b). a) Les polygones en trait plein ($\mathbf{v}$, « piqué » en son i -ème sommet) et pointillés ($\tilde{\mathbf{v}}$, « plat » en son i -ème sommet) mènent à deux barycentres identiques pour le i -ème sommet ($v_i^* = \tilde{v}_i^*$), donc à la même valeur de $|v_i - v_i^*|$, b) Deux polygones triangulaires $\mathbf{v}$ et $\tilde{\mathbf{v}}$, l'un aplati et l'autre équilatéral, possèdent le même cercle circonscrit. Comme $\mathbf{t}_1 = \mathbf{v}$ et $\tilde{\mathbf{t}}_1 = \tilde{\mathbf{v}}$, leurs cercles circonscrits (donc leurs rayons) sont identiques.

associée à $\mathcal{R}$ est de la forme $p(\mathbf{v}) \propto \exp[-\mathcal{R}(\mathbf{v})/T]$, et que l'estimateur du maximum *a posteriori* (MAP), noté $\hat{\mathbf{v}}_{\text{MAP}}$ ou $\hat{\mathbf{v}}$, minimise le critère pénalisé

$$\mathcal{J}(\mathbf{v}) = \mathcal{K}(\mathbf{v}) + \lambda \mathcal{R}(\mathbf{v}), \quad (\text{IV.18})$$

où l'hyperparamètre $\lambda > 0$ établit un compromis entre le terme d'attache aux données et la douceur du polygone.

Définition des courbures. La régularisation définie ci-dessus est de type « markovienne », au sens où l'énergie $\mathcal{R}(\mathbf{v})$ est la somme de quantités locales aux sommets v_i . En fait, en choisissant des courbures approchées $\kappa_i(\mathbf{v})$ qui ne dépendent que des sommets v_{i-1}, v_i, v_{i+1} , la loi *a priori* $p(\mathbf{v})$ définit une chaîne de Markov. En effet, la loi conditionnelle $p(v_i | v_j, j \neq i)$ est identique à

$$p(v_i | v_j, j \sim i) \propto \exp\left[-\{\phi(\kappa_{i-1}(\mathbf{v})) + \phi(\kappa_i(\mathbf{v})) + \phi(\kappa_{i+1}(\mathbf{v}))\}/T\right],$$

où « $\sim$ » représente le système de voisinage d'ordre 1. Différents choix sont possibles pour $\kappa_i(\mathbf{v})$. Le plus simple est de faire intervenir la distance d'un sommet au barycentre de ses voisins, *i.e.*,

$$\kappa_i(\mathbf{v}) = |v_i - v_i^*|, \text{ où } v_i^* = (v_{i-1} + v_{i+1})/2. \quad (\text{IV.19})$$

Ce choix a une interprétation statistique simple : les lois *a priori* conditionnelles $p(v_i | v_j, j \neq i)$ sont centrées en les barycentres v_i^* (Mohammad-Djafari, 1997). De plus, le choix d'une fonction ϕ quadratique mène à des lois gaussiennes. En revanche, ce choix présente l'inconvénient de dépendre de la taille du triangle $\mathbf{t}_i = (v_{i-1}, v_i, v_{i+1})$. Deux polygones homothétiques sont donc pénalisés de façon différente alors que leur forme est identique. Bulpitt et Efford (1996) pallient ce problème en normalisant dans (IV.19) la distance $|v_i - v_i^*|$ par la taille du voisinage de v_i . Ces auteurs envisagent également une pénalisation qui porte sur la hauteur du triangle $\mathbf{t}_i$ issue de v_i , avec une normalisation analogue sur les distances. Ce n'est pas la solution que nous adoptons, car la taille du voisinage de v_i est une notion arbitraire. Nous préférons utiliser une forme simple et directe pour $\kappa_i(\mathbf{v})$.

L'idée naturelle est alors de définir la valeur absolue de la courbure en v_i comme l'inverse du rayon du cercle circonscrit au triangle t_i . Mais ce choix n'est pas satisfaisant, car il mène aux mêmes problèmes que précédemment, comme illustré sur la figure IV.4 (b). Nous proposons donc de régulariser directement les angles $A_i(\mathbf{v}) \in (0, \pi)$ en les sommets du polygone :

$$\kappa_i(\mathbf{v}) = 1 + \cos A_i(\mathbf{v}). \quad (\text{IV.20})$$

Nous ne pénalisons pas les sommets en lesquels l'angle est plat, *i.e.*, $\cos A_i(\mathbf{v}) = -1$, et pénalisons au maximum les sommets aigus, tels que $\cos A_i(\mathbf{v}) \approx 1$. C'est un choix très proche de l'énergie qui intervient dans les champs de Arak (Arak et Surgailis, 1989) et de celle qu'utilisent Hanson et al. (1997).

Critère global. Nous résumons l'étude précédente en exprimant la fonctionnelle régularisante :

$$\mathcal{R}(\mathbf{v}) = \sum_{i=1}^n \phi(1 + \cos(A_i(\mathbf{v}))), \quad (\text{IV.21})$$

où chaque terme $A_i(\mathbf{v})$ est l'angle en le i -ème sommet de $\mathbf{v}$. La fonction ϕ est choisie paire et croissante sur $\mathbb{R}_+$. Nous avons envisagé plusieurs choix, et notamment des potentiels non convexes, comme les fonctions quadratiques en 0 et constantes à l'infini (Geman et McClure, 1987), déjà évoquées au chapitre II (§ II.2.2), de façon à ne pas exclure les grandes déformations d'un sommet. L'expérience montre que de tels potentiels engendrent une augmentation nette du nombre de minima locaux de $\mathcal{J}$. Par conséquent, nous optons pour un potentiel convexe simple : $\phi(t) = t^2$ (fonction $\mathcal{L}_2$).

IV.3 Optimisation du critère

Nous avons vu précédemment que la reconstruction de polygones se ramène à la minimisation contrainte (sur $\mathbb{A}_n$) de l'énergie *a posteriori* $\mathcal{J}(v_1, \dots, v_n)$, critère généralement non convexe (voir § IV.A). D'une façon générale, il existe deux grandes classes de méthodes d'optimisation de fonctionnelles non convexes, déjà évoquées au chapitre II :

- Les méthodes stochastiques de type recuit simulé (Geman et Geman, 1984), pour lesquelles la convergence est prouvée, mais s'avère extrêmement lente en pratique.
- Les méthodes itératives déterministes d'optimisation locale.

Pour simplifier la mise en œuvre, nous retenons la deuxième catégorie de méthodes. Ces méthodes produisent toutes des minima locaux de $\mathcal{J}$, et sont donc très sensibles à l'initialisation. Pour atteindre le minimum global de $\mathcal{J}$, il faut que l'initialisation appartienne à son bassin d'attraction (voir § IV.3.3). Nous étudions maintenant ces méthodes locales, et notamment celles du premier ordre.

IV.3.1 Techniques locales d'optimisation

Tout d'abord, nous mettons l'accent sur le fait que la plupart de ces techniques nécessitent la connaissance du gradient de $\mathcal{J}$. En fait, l'algorithme du simplexe (Nelder et Mead, 1965) permet l'optimisation locale sans cette information. C'est l'algorithme utilisé par Milanfar et al. (1994) pour un problème d'optimisation similaire au nôtre. Il converge rapidement pour leur application, car le nombre de sommets des polygones manipulés est faible. Ce n'est pas notre cas (ce le sera d'autant moins en 3D), et un tel algorithme n'est pas envisageable. Pour pouvoir utiliser les méthodes du premier ordre, l'évaluation du gradient de $\mathcal{J}$ est essentielle. À cet effet, nous proposons au § IV.4.4 une méthode originale de calcul exact de $\nabla \mathcal{J}$. Nous considérons ci-dessous deux grands types de méthodes du premier ordre.

```

/* Initialisation  $\mathbf{v}^{(0)}$  */
Reconstruction d'un contour  $\mathcal{C}_{\hat{\mathbf{x}}}$  paramétré « globalement » (cf. chapitre III).
Approximation de  $\mathcal{C}_{\hat{\mathbf{x}}}$  par un polygone  $\mathcal{P}_{\hat{\mathbf{x}}}$  à  $n$  sommets.
Faire  $\mathcal{P}^{(0)} = \mathcal{P}_{\hat{\mathbf{x}}}$ , i.e.,  $\mathbf{v}^{(0)} = \mathbf{v}_{\hat{\mathbf{x}}}$ .
Pour  $k = 1, \dots, K$ ,
  /* Définir un ordre de parcours des sommets */
  Pour  $j = 1, \dots, n$ ,
    Résoudre (IV.25) avec  $i = \text{ordre}(j)$ .
    Mettre à jour  $v_i^{(k)}$  selon (IV.24).
  Fin Pour.
  Former  $\mathbf{v}^{(k)} = \{v_1^{(k)}, \dots, v_n^{(k)}\}$ .
Fin Pour.

```

Tableau IV.1 – Méthode locale de descente par blocs (sommet par sommet). L'ordre du parcours des sommets est une permutation de $\{1, \dots, n\}$, choisie aléatoirement en pratique.

Méthode de descente sur $\mathbf{v}$. L'algorithme du gradient (non projeté) optimise $\mathcal{J}$ directement sur $\mathbb{R}^{2n}$, i.e., sur tous les sommets simultanément. Sous sa forme la plus simple, il s'exprime par :

$$\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} - s_k \nabla \mathcal{J}(\mathbf{v}^{(k)}), \quad \text{où } s_k = \arg \min_{s \in \mathbb{R}_+} \mathcal{J}(\mathbf{v}^{(k)} - s \nabla \mathcal{J}(\mathbf{v}^{(k)})). \quad (\text{IV.22})$$

La quantité s_k représente le pas de descente. Nous renvoyons par ailleurs le lecteur au § II.3 pour la description de l'algorithme du gradient projeté pour la minimisation sous contraintes convexes. La raison pour laquelle nous n'effectuons pas de minimisation contrainte (pour garantir l'appartenance de $\mathbf{v}^{(k)}$ à $\mathbb{A}_n$) est que les contraintes ne s'expriment pas simplement en fonction des variables. De plus, nous n'avons pas d'information sur la convexité de $\mathbb{A}_n$, ce qui exclut l'utilisation d'un algorithme « projeté sur l'ensemble des contraintes ».

Méthode de descente par blocs (GS). Une alternative au schéma précédent est l'algorithme de descente par blocs, où les blocs correspondent aux sommets v_i de $\mathbf{v}$ (variables de $\mathbb{R}^2$). Dans un premier temps, nous décrivons l'algorithme *exact* de relaxation par blocs, puis nous l'adaptions au paragraphe suivant. Cet algorithme consiste à réaliser séquentiellement la minimisation de $\mathcal{J}(\mathbf{v})$ par rapport à chaque sommet v_i (en considérant tous les autres fixés). Ses motivations pour notre problème sont de deux ordres :

- La faible charge calculatoire associée à la remise à jour du critère $\mathcal{J}$ suite au déplacement d'*un seul* sommet (ce point sera détaillé au § IV.4.3).
- La gestion des auto-intersections des contours (développée au § IV.3.2).

Nous décrivons à présent le schéma général de l'algorithme de relaxation par bloc, appelé Gauss-Seidel (GS) en analyse numérique. Il s'agit d'un algorithme itératif (K itérations), où à chaque itération k , tous les sommets de $\mathbf{v}^{(k-1)}$ sont parcourus et mis à jour de la façon suivante :

$$\forall i \in \{1, \dots, n\}, \quad v_i^{(k)} = \arg \min_{v_i} \{\mathcal{J}(\mathbf{v}) = \mathcal{J}(v_i; v_j, j \neq i)\}. \quad (\text{IV.23})$$

Dans le schéma (IV.23), l'ordre du parcours des sommets est fixé ($i = 1, 2, \dots, n$), mais il est souvent plus efficace de ne pas décrire successivement les sommets voisins (voir le tableau IV.1).

```

/* État initial : nous disposons de  $\mathbf{v}^{(k-1)}$  et  $\mathcal{A}(\mathbf{v}^{(k-1)})$  */
Faire  $\mathbf{p} = \mathcal{A}(\mathbf{v}^{(k-1)})$ .
Pour  $i = 1, \dots, n$ ,
    Estimer  $v_i^{(k)}$  à partir de  $v_1^{(k)}, \dots, v_{i-1}^{(k)}, v_i^{(k-1)}, \dots, v_n^{(k-1)}$  selon (IV.24).
    /* Mise à jour de  $\mathbf{p}$  */
    Renseigner  $\mathbf{p} = \mathbf{p} + \varepsilon_i^{(k)} \mathcal{A}(\mathbf{t}_i^{(k)}) - \varepsilon_i^{(k-1)} \mathcal{A}(\mathbf{t}_i^{(k-1)})$ .
Fin Pour.
Calculer directement  $\mathcal{A}(\mathbf{v}^{(k)})$  par l'algorithme IV.4 (page 72).
/* Comparaison */
Si  $\|\mathcal{A}(\mathbf{v}^{(k)}) - \mathbf{p}\| > \varepsilon$ ,
     $\mathbf{v}^{(k)} = \{v_i^{(k)}, i = 1, \dots, n\}$  n'est pas acceptable.
Fin Si.

```

Tableau IV.2 – Test d'acceptabilité à l'itération k pour la procédure GS (où par lisibilité, nous décrivons successivement $v_1, \dots, v_n$). Ce test s'appuie sur la mise à jour des projections de $\mathbf{v}^{(k-1)}$ (quantité $\mathbf{p}$) à chaque fois qu'un sommet $v_i^{(k-1)}$ est déplacé (en $v_i^{(k)}$). La mise à jour consiste à projeter deux triangles $\mathbf{t}_i^{(k-1)}$ et $\mathbf{t}_i^{(k)}$ (voir § IV.4.3).

En pratique, nous ne réalisons pas les minimisations (IV.23) de façon exacte, mais nous optons pour une stratégie *hybride*, en remplaçant les n optimisations (IV.23) par n minimisations de ligne (sur les sommets v_i) des fonctionnelles $v_i \rightarrow \mathcal{J}(v_i; v_j, j \neq i)$:

$$\forall i \in \{1, \dots, n\}, \quad v_i^{(k)} = v_i^{(k-1)} - s_i^{(k)} \nabla \mathcal{J}(v_i^{(k-1)}; v_j, j \neq i), \quad (\text{IV.24})$$

où $\nabla \mathcal{J}(v_i; v_j, j \neq i)$ représente la dérivée partielle $\partial \mathcal{J}(\mathbf{v}) / \partial v_i \in \mathbb{R}^2$, et les pas de descente $s_i^{(k)}$ sont obtenus par :

$$s_i^{(k)} = \arg \min_{s \in \mathbb{R}_+} \mathcal{J}(v_i^{(k-1)} - s \nabla \mathcal{J}(v_i^{(k-1)}; v_j, j \neq i); v_j, j \neq i). \quad (\text{IV.25})$$

L'algorithme utilisé est en fait un algorithme de descente car la suite des itérées $\mathcal{J}(\mathbf{v}^{(k)})$ (résultant de (IV.24)) est décroissante. Il est finalement résumé par le tableau IV.1, dans lequel l'étape d'initialisation est incluse, et l'ordre de parcours des sommets est une permutation aléatoire de $\{1, \dots, n\}$.

Dans la suite de cette section, nous choisissons une stratégie particulière, en insistant sur deux aspects précis : le caractère acceptable des estimées et le choix de son initialisation.

IV.3.2 Stratégie retenue

Indépendamment de la charge calculatoire, le choix de l'algorithme dépend de la forme du critère $\mathcal{J}$. Plus précisément, les méthodes de descente par blocs sont plus efficaces pour les optimisations dans les « vallées plates » de $\mathcal{J}$, *i.e.*, sur les domaines où les variations de $\mathcal{J}$ sont faibles. Bien sûr, nous ne disposons pas de cette information en général. Un compromis intéressant est de procéder en deux temps : utiliser une méthode de descente « globale » (sur $\mathbf{v}$) pendant un certain nombre d'itérations, puis une méthode par blocs pour raffiner la solution intermédiaire. Pour notre application, nous ne mettons pas en avant la charge calculatoire des algorithmes, car elle est peu élevée⁴. En revanche, la gestion du caractère acceptable des polygones reconstruits (au sens de la définition (IV.2) ; voir aussi la figure IV.1 page 58) est un point qui nous semble essentiel. En d'autres termes, il s'agit de tester simplement la propriété : $\mathbf{v}^{(k)} \in \mathbb{A}_n, \forall k$.

⁴Le coût de la méthode par blocs est le plus élevé, mais tout à fait acceptable en 2D : voir les développements du § IV.4.3 pour la mise à jour des projections.

Bien sûr, nous excluons la recherche systématique des intersections deux à deux des arêtes de $\mathbf{v}^{(k)}$, cette procédure étant bien trop lourde. Une façon naturelle de procéder est d'utiliser une pixellisation de l'objet, comme le font Battle et al. (1998) dans un contexte 3D⁵. Ils mettent à jour, pour chaque pixel, les arêtes qui l'intersectent, et détectent une auto-intersection si deux arêtes non adjacentes intersectent le même pixel. Cette stratégie est efficace, bien qu'elle ne garantisse pas en théorie le caractère acceptable ou non acceptable des polygones. Pour notre part, nous préférons mettre à profit le calcul des projections de $\mathbf{v}$ et leurs mises à jour pour contrôler le caractère acceptable de $\mathbf{v}$. Cela nous mène à des tests différents suivant la stratégie d'optimisation employée. Considérons la k -ème itération de l'algorithme d'optimisation de $\mathcal{J}$, qui estime $\mathbf{v}^{(k)}$ à partir de $\mathbf{v}^{(k-1)}$.

- Pour la stratégie GS, nous définissons la quantité intermédiaire $\mathbf{p} = \mathcal{A}(\mathbf{v}^{(k-1)})$, que nous mettons à jour pour chaque sommet déplacé par la procédure détaillée au § IV.4.3. Une fois l'itération k terminée, les n sommets $v_1^{(k)}, \dots, v_n^{(k)}$ ont été ré-estimés, et nous recalculons directement les projections $\mathcal{A}(\mathbf{v}^{(k)})$, puis nous comparons ce résultat avec $\mathbf{p}$. Cette procédure est résumée par le tableau IV.2. Le test $\mathcal{A}(\mathbf{v}^{(k)}) = \mathbf{p}$ est une condition nécessaire mais *a priori* non suffisante pour garantir le caractère acceptable de $\mathbf{v}^{(k)}$. En pratique, il s'avère révélateur. Néanmoins, s'il est négatif, il n'indique pas de quelle façon modifier les sommets de $\mathbf{v}^{(k-1)}$ pour rester dans le domaine $\mathbb{A}_n$ des polygones acceptables.
- Pour l'algorithme du gradient, une telle procédure n'est pas envisageable car nous déplaçons globalement l'ensemble des sommets. Toutefois, nous proposons une condition nécessaire d'acceptabilité de $\mathbf{v}^{(k)}$, qui repose sur les éléments suivants. Nous pouvons montrer que si un polygone $\mathbf{v}$ n'est pas acceptable, les algorithmes de projections mènent néanmoins au calcul exact de $\mathcal{A}(\mathbf{v})$, puis de $\mathcal{J}(\mathbf{v})$. Par contre, l'algorithme de calcul de $\nabla \mathcal{J}(\mathbf{v})$ ne s'étend pas au cas où $\mathbf{v} \notin \mathbb{A}_n$. Un test d'acceptabilité de l'estimée $\mathbf{v}^{(k)}$ consiste donc à comparer les dérivées partielles $\partial \mathcal{J}(\mathbf{v}^{(k)}) / \partial v_i$ à leurs approximations par différences finies.

Finalement, la stratégie sommet par sommet apparaît la plus adaptée, bien que son coût algorithmique soit plus élevé.

IV.3.3 Initialisation et critère d'arrêt

Pour les deux algorithmes, l'initialisation est un point clé car elle conditionne fortement le résultat de l'optimisation. En effet, pour obtenir le minimum global $\mathbf{v}^*$ de $\mathcal{J}$, il faut que l'initialisation soit suffisamment bonne, *i.e.*, dans le bassin d'attraction de $\mathbf{v}^*$. C'est un point que nous évoquons à nouveau au chapitre VI à l'aide de simulations numériques. Pour cela, nous montrons et quantifions la qualité des reconstructions obtenues sur un même jeu de données, à partir d'initialisations différentes. Nous considérons des ellipses, superquadriques et surfaces harmoniques, obtenues par la méthode de reconstruction présentée au chapitre III. Nous verrons que ces dernières fournissent les meilleurs résultats, au détriment certes de la rapidité de calcul.

Enfin, nous précisons les critères d'arrêt des algorithmes d'optimisation. Dans les deux cas, ce sont des algorithmes de descente, et nous arrêtons leur déroulement à l'itérée $k < K$ si

$$\mathcal{J}(\mathbf{v}^{(k-1)}) - \mathcal{J}(\mathbf{v}^{(k)}) < \varepsilon, \quad (\text{IV.26})$$

où ε est un seuil arbitraire. D'autre part, le nombre K d'itérations est fixé empiriquement (typiquement, $K \approx 50$ pour des polygones à $n \approx 100$ sommets).

⁵En fait, ils utilisent la pixellisation pour calculer un critère pénalisé et son gradient. Puis, ils en tirent partie pour tester l'acceptabilité de l'objet.

IV.3.4 Conclusions

Nous résumons brièvement l'étude précédente. Les deux méthodes locales proposées fournissent des résultats comparables avec des coûts numériques faibles. L'approche par blocs (sommet par sommet) apparaît la plus adéquate car elle permet à chaque itération de vérifier que l'estimée $\mathbf{v}^{(k)}$ ne s'intersecte pas. Pour les deux méthodes, l'initialisation est un élément important, car elle conditionne le résultat de l'optimisation. Nous optons pour une reconstruction préalable par une méthode plus simple : l'estimation par moindres carrés des paramètres d'un modèle harmonique (voir le chapitre III).

Dans la partie suivante, nous développons les étapes nécessaires à la mise en œuvre de ces algorithmes. Il s'agit du calcul et de la mise à jour des projections suite au déplacement de l'un des sommets, et de l'évaluation du gradient de $\mathcal{J}$. Cette partie est relativement technique, mais essentielle à la minimisation de $\mathcal{J}$. Néanmoins, les éléments les plus techniques sont développés en annexe C.

IV.4 Étude des projections

Nous nous intéressons maintenant à l'évaluation directe des projections de $\mathbf{v}$ (§ IV.4.1 et IV.4.2), et à leur remise à jour suite à des déplacements de sommets (§ IV.4.3 et IV.4.4). Nous proposons des algorithmes originaux à cet effet. Ils sont essentiels car ils interviennent constamment au cours du processus de reconstruction.

Tout d'abord, nous remarquons que les polygones ont des projections de forme très simple : elles sont affines par morceaux. Considérons une seule projection $p(t)$ de $\mathbf{v}$. Il est envisageable de définir $p(t)$ sur $\mathbb{R}$ à partir uniquement des valeurs de p en les projetés des sommets v_i . Cette stratégie pose en fait problème dans certains cas ; si l'une des arêtes adjacentes à v_i est parallèle à la direction de projection, $p(t)$ n'est pas continue en le projeté de v_i . Ces remarques nous mènent à évaluer chaque valeur projetée $p_\theta(t)$ de façon indépendante. Dans les deux paragraphes suivants, nous décrivons deux classes de méthodes d'évaluation de $p_\theta(t)$ (à t et θ fixés).

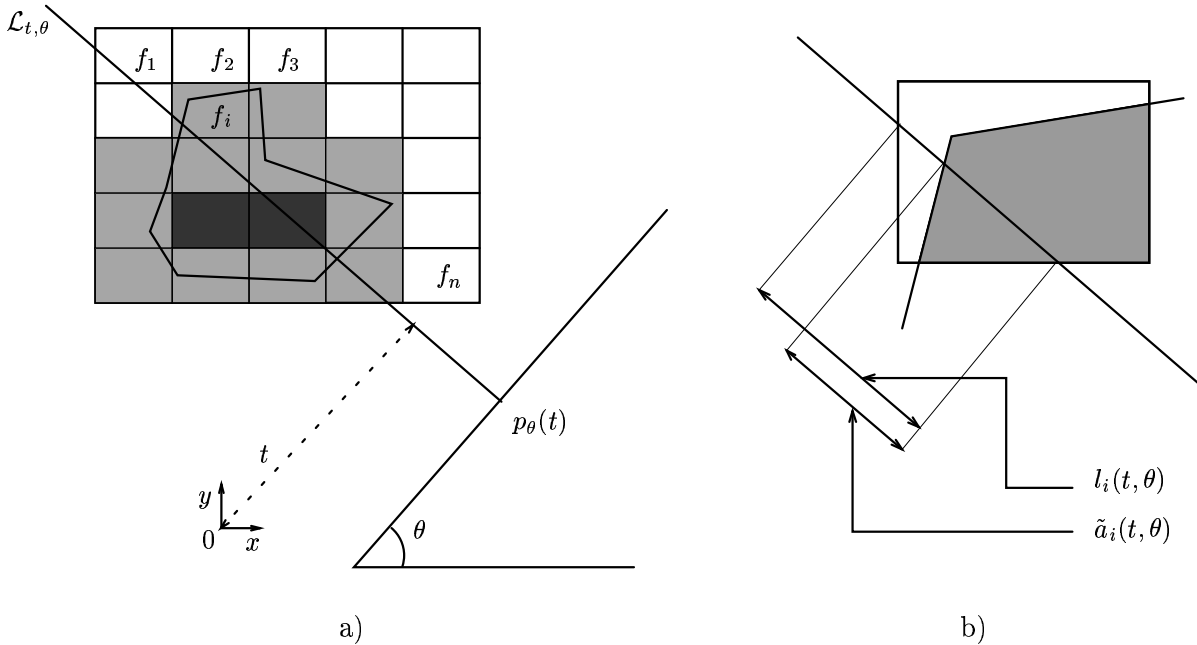


Figure IV.5 – Évaluation *approchée* d'une projection d'un polygone par discrétisation en pixels. a) Mesure de $p_\theta(t)$. b) Contribution du i -ème pixel à cette mesure (pixel frontière).

IV.4.1 Calcul approché

La première approche consiste à discrétiser la scène en un ensemble de pixels $\mathbf{f}$, puis de projeter cette scène approchée par l'opérateur linéaire $\mathbf{p} = \mathbf{A}\mathbf{f}$, déjà introduit au chapitre II. C'est l'approche choisie par Battle et al. (1998) pour des applications en résonance magnétique nucléaire, avec un modèle direct qui fait intervenir un faisceau de rayons par pixel détecteur. Dans leurs travaux, le vecteur $\mathbf{f}$ n'est pas binaire comme au chapitre II. Ils assignent aux pixels f_i les valeurs suivantes :

- 0 pour les pixels entièrement à l'extérieur de $\mathbf{v}$.
- 1 pour les pixels entièrement à l'intérieur de $\mathbf{v}$.
- La proportion de l'aire de $\{i\} \cap \mathbf{v}$ par rapport à celle de $\{i\}$ pour les pixels frontières (par exemple $f_i = 0,75$ pour le pixel frontière repéré sur la figure IV.5 (a)).

Sur la figure IV.5 (a), nous représentons un polygone $\mathbf{v}$ et sa discrétisation $\mathbf{f}$, puis le rayon $\mathcal{L}_{t,\theta}$. La contribution d'un pixel frontière f_i à la projection $p_\theta(t)$ est la quantité $l_i(t, \theta) f_i$. C'est une approximation de la véritable contribution $\tilde{a}_i(t, \theta)$, avec les notations de la figure IV.5 (b).

Cette approche présente l'avantage d'être plus facilement adaptable à des modèles mixtes, où l'intérieur et l'extérieur de $\mathbf{v}$ ne sont pas uniformes. En revanche, elle s'avère relativement technique pour ce qui concerne l'évaluation des valeurs des pixels frontières. Dans ce qui suit, nous proposons un calcul exact et rapide des projections, basé sur les recherches d'intersections rayons-arêtes.

IV.4.2 Calcul direct

Les algorithmes décrits ci-après s'inspirent de (Arvo, 1991). Comme nous l'avons vu au chapitre III (§ III.1), l'évaluation de la projection d'un polygone $\mathbf{v}$ s'appuie sur les calculs d'intersection entre le contour $\mathbf{v}$ et les rayons de projection $\mathcal{L}_{t,\theta}$. Nous fixons donc un tel rayon, noté $\mathcal{L}$, passant par la source S , orienté par $\mathbf{w} = \mathbf{w}_{\theta-\pi/2}$ et arrivant en un pixel détecteur t . Le calcul de $p_\theta(t)$ dépend des intersections entrantes et sortantes entre $\mathcal{L}$ et les arêtes de $\mathbf{v}$. Deux situations peuvent se produire, et sont décrites ci-dessous.

Le cas régulier (figure IV.6). Aucun sommet v_i n'appartient à $\mathcal{L}$. Alors le nombre d'intersections $\mathcal{L} \cap (v_i, v_{i+1})$ est pair. Notons I_j , $j = 1, \dots, 2k$ leurs positions, et l_j leurs distances à S , soit

$$I_j(l_j) = S + l_j \mathbf{w}. \quad (\text{IV.27})$$

Les distances $l_j > 0$ sont triées par ordre croissant, nous avons donc :

$$p_\theta(t) = \int_{\mathcal{L} \cap \mathbf{v}} dl = \sum_{j=1}^k d(I_{2j-1}, I_{2j}) = \sum_{j=1}^k (l_{2j} - l_{2j-1}). \quad (\text{IV.28})$$

Les positions I_{2j-1} correspondent aux points d'entrée du rayon dans $\mathbf{v}$, et les positions I_{2j} aux points de sortie.

Le cas singulier (figure IV.7). Au moins un sommet v_i appartient à $\mathcal{L}$. Dans le cas le plus fréquent, il n'y en a qu'un, et le nombre d'intersections entre $\mathcal{L}$ et les arêtes (v_i, v_{i+1}) est impair. Il n'existe pas de façon immédiate de traiter ce cas, *i.e.*, de déterminer les positions I_j correspondant à des intersections singulières. Nous préconisons la recherche des intersections de $\mathcal{L}_{t+\delta t, \theta}$ avec les arêtes (v_j, v_{j+1}) pour en déduire par «continuité» les positions I_j qui sont à exclure. Cette étape n'intervient que pour inspecter les positions I_j singulières mais l'évaluation de $p_\theta(t)$ est exacte, car repose sur les intersections régulières de $\mathcal{L} = \mathcal{L}_{t,\theta}$ et les arêtes (v_j, v_{j+1}) .

Pour une plus grande clarté de l'exposé, nous n'évoquons dans la suite que les cas de polygones réguliers. Nous faisons une distinction entre les polygones réguliers convexes et les polygones réguliers non convexes, car il existe des algorithmes rapides pour les premiers.

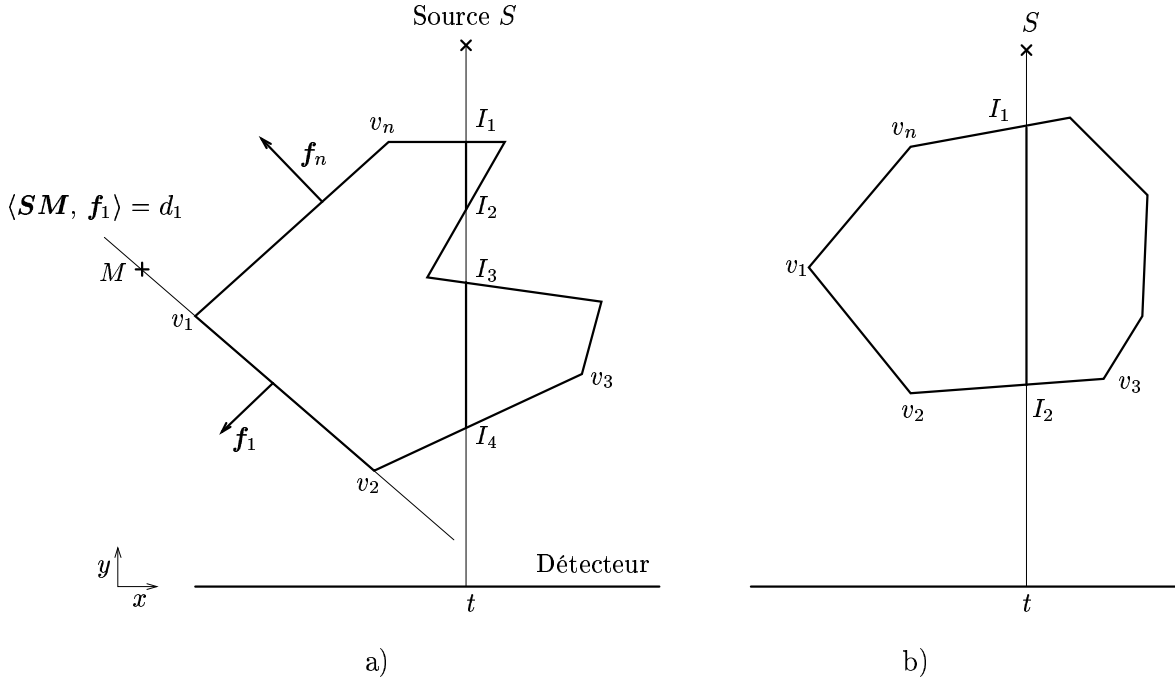


Figure IV.6 – Évaluation d’une projection (projection d’angle $\theta = 0$) du polygone v dans le cas régulier : le nombre d’intersections I_j entre le rayon $\mathcal{L}$ et v est pair. a) Pour un polygone quelconque, $I_1, I_3, \dots$ correspondent aux entrées du rayon dans l’objet, $I_2, I_4, \dots$ correspondent aux sorties. b) Pour un polygone convexe, il n’y a que deux intersections I_1 et I_2 , et la valeur de la projection est égale à la distance entre ces points.

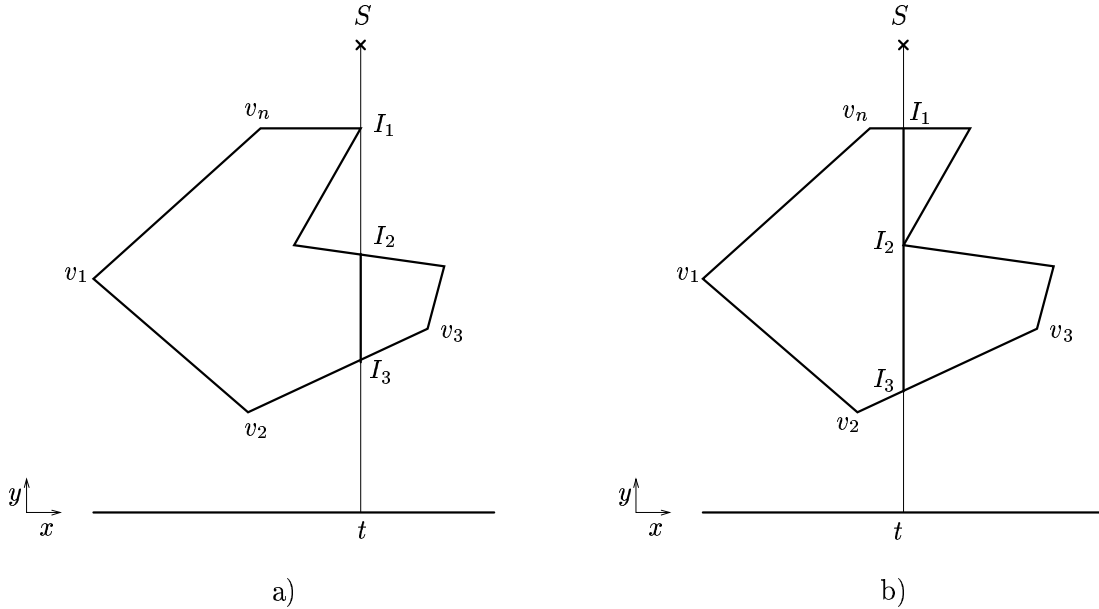


Figure IV.7 – Évaluation d’une projection (projection d’angle $\theta = 0$) du polygone v dans le cas singulier, où le rayon de projection passe exactement sur un (ou plusieurs) sommet de v . S’il n’y a qu’une singularité, le nombre d’intersections I_j entre $\mathcal{L}$ et v est impair. L’évaluation de $p(t)$ à partir des positions I_j est indéterminée (comme illustré en (a) et (b)), et requiert des calculs complémentaires.

IV.4.2.1 Cas convexe

Nous utilisons une propriété fondamentale de la convexité des polygones. Elle nous permet d'éviter les calculs d'intersections entre $\mathcal{L}$ et les n arêtes (v_j, v_{j+1}) , et de nous limiter uniquement aux intersections entre $\mathcal{L}$ et les droites qui les supportent. En définissant pour tout j ,

- la normale sortante $\mathbf{f}_j$ à (v_j, v_{j+1}) ,
- la distance algébrique d_j de S à $\text{Supp}(v_j, v_{j+1})$,

notions illustrées en figure IV.6, l'équation de la droite qui supporte (v_j, v_{j+1}) est $\langle \mathbf{S}\mathbf{M}, \mathbf{f}_j \rangle = d_j$, et le j -ème demi-plan « intérieur » à $\mathbf{v}$ est défini par :

$$\mathbb{I}_j = \{M \in \mathbb{R}^2, \langle \mathbf{S}\mathbf{M}, \mathbf{f}_j \rangle \leq d_j\}. \quad (\text{IV.29})$$

La convexité de $\mathbf{v}$ implique alors

$$\mathbf{v} = \bigcap_{j=1}^n \mathbb{I}_j. \quad (\text{IV.30})$$

Dans le cas où $\mathcal{L}$ intersecte $\mathbf{v}$ mais n'est confondu avec aucune droite $\text{Supp}(v_j, v_{j+1})$, il n'y a que deux intersections $\underline{l}$ et $\bar{l}$ qui interviennent dans le calcul (IV.28), et $p_\theta(t) = \bar{l} - \underline{l}$ (voir Arvo, 1991). Plus précisément, nous notons $I_j(l_j) = \mathcal{L} \cap \text{Supp}(v_j, v_{j+1})$. D'après (IV.30),

$$\underline{l} = \max \{l_j, \langle \mathbf{f}_j, \mathbf{w} \rangle < 0\}, \quad (\text{IV.31})$$

$$\bar{l} = \min \{l_j, \langle \mathbf{f}_j, \mathbf{w} \rangle > 0\}, \quad (\text{IV.32})$$

où $l_j = d_j / \langle \mathbf{f}_j, \mathbf{w} \rangle$. D'où un moyen simple d'évaluer $p_\theta(t)$. Enfin, dans le cas limite où une arête (v_j, v_{j+1}) est incluse dans $\mathcal{L}$, *i.e.*, $d_j = 0$, nous ne tenons pas compte de cette arête. En effet, il y a une infinité d'intersections entre $\mathcal{L}$ et (v_j, v_{j+1}) , et elles n'influent pas sur le calcul de la projection. L'évaluation de $p_\theta(t)$ est finalement détaillée par le tableau IV.3.

```

Faire  $\underline{l} = -\infty, \bar{l} = \infty$ .
Pour tout  $j \in \{1, \dots, n\}$ ,    /* Inspection de l'arête  $(v_j, v_{j+1})$  */
    Déterminer les caractéristiques  $(\mathbf{f}_j, d_j)$  de l'arête  $(v_j, v_{j+1})$ .
    Si  $|\langle \mathbf{f}_j, \mathbf{w} \rangle| < \varepsilon$ ,      /*  $\mathcal{L}$  est parallèle à  $(v_j, v_{j+1})$  */
        Si  $d_j < 0$ , renvoyer 0. Fin Si.    /*  $\mathcal{L}$  est à l'extérieur de  $\mathbf{v}$  */
    Sinon,
        Calculer  $l_j = d_j / \langle \mathbf{f}_j, \mathbf{w} \rangle$ .
        Si  $\langle \mathbf{f}_j, \mathbf{w} \rangle > 0$ , faire  $\bar{l} = \min(\bar{l}, l_j)$ .    /* Intersection sortante */
        Sinon, faire  $\underline{l} = \max(\underline{l}, l_j)$ .
        Fin Si.
    Fin Si.
    Si  $\underline{l} \geq \bar{l}$ , renvoyer 0. Fin Si.
Fin Pour.
Renvoyer  $\bar{l} - \underline{l}$ .

```

Tableau IV.3 – Algorithme de projection d'un polygone convexe. Le calcul de $p_\theta(t)$ est basé sur les intersections entre $\mathcal{L}$ et les droites qui supportent les arêtes.

IV.4.2.2 Cas non convexe

Le calcul de $p_\theta(t)$ nécessite alors les évaluations explicites des intersections rayons-arêtes. Pour une arête (v_i, v_{i+1}) fixée, il s'agit de rechercher les positions $M \in \mathbb{R}^2$ telles que $\mathbf{S}\mathbf{M} = l\mathbf{w}$ ($l > 0$) et $\mathbf{v}_i\mathbf{M} = b\mathbf{v}_i\mathbf{v}_{i+1}$, où $b \in (0, 1)$ indique que l'intersection appartient bien à l'arête.

Proposition 4 *Les expressions de b et l sont données par*

$$b = -\langle \mathbf{S}\mathbf{v}_i, \mathbf{w}^\perp \rangle / \langle \mathbf{f}_i, \mathbf{w} \rangle, \quad (\text{IV.33})$$

$$l = \langle \mathbf{S}\mathbf{v}_i, \mathbf{f}_i \rangle / \langle \mathbf{f}_i, \mathbf{w} \rangle, \quad (\text{IV.34})$$

où $\mathbf{w}^\perp$ est le vecteur unitaire directement orthogonal à $\mathbf{w}$. Le cas $\langle \mathbf{f}_i, \mathbf{w} \rangle = 0$ correspond à une direction $\mathcal{L}$ parallèle à (v_i, v_{i+1}) .

L'algorithme qui implémente la formulation (IV.28) est finalement spécifié par le tableau IV.4.

```

Faire  $k = 0$ . /* Nombre d'intersections */
Pour tout  $j \in \{1, \dots, n\}$ , /* Évaluation de  $I_j(l_k) = \mathcal{L} \cap (v_j, v_{j+1})$  */
    Si  $|\langle \mathbf{f}_j, \mathbf{w} \rangle| > \varepsilon$ , /*  $\mathcal{L}$  n'est pas parallèle à  $(v_j, v_{j+1})$  */
        Calculer  $b_j$  par (IV.33).
        Si  $b_j \in (0, 1)$ , faire  $k = k + 1$  et calculer  $l_k$  par (IV.34).
    Fin Si.
Fin Pour.
Si  $k$  est impair, /* Détection d'une intersection singulière */
    Trier la séquence  $\{l_j, j = 1, \dots, k\}$ .
    /* Élimination des points doubles */
    Pour  $j \in \{1, \dots, k-1\}$ , Si  $l_j = l_{j+1}$ ,
        Enlever  $I_{j+1}$  de la liste des points d'intersection.
        Faire  $k = k - 1$ .
    Fin Si, Fin Pour.
Fin Si.
Si  $k$  est pair, faire  $k = k/2$ , trier  $\{l_j, j = 1, \dots, 2k\}$ , puis calculer  $p_\theta(t)$  par (IV.28).
Sinon,
    Rechercher les intersections  $\mathcal{L} \cap (v_j, v_{j+1})$  avec  $S$  déplacé en  $S + \delta S$ .
    En déduire les  $p$  intersections ( $p$  impair)  $I_j(l_j)$  à éliminer. Faire  $k = (k - p)/2$ .
    Calculer  $p_\theta(t)$  par (IV.28).
Fin Si.

```

Tableau IV.4 – Algorithme de projection d'un polygone non convexe. Pour le calcul de $p_\theta(t)$, la recherche de toutes les intersections $I_j(l_j)$ du rayon $\mathcal{L}$ et des arêtes (v_j, v_{j+1}) est nécessaire. Nous calculons et trions la liste des l_j .

IV.4.3 Mise à jour des projections

Après avoir étudié le calcul direct des projections d'un polygone, nous considérons leur mise à jour suite au déplacement d'un seul sommet par rapport à une configuration initiale. Cette question est essentielle pour la mise en œuvre de l'algorithme de descente par blocs que nous prônons. Nous montrons que cette opération découle directement de la décomposition d'un polygone en triangles, et qu'elle peut être gérée de façon simple, en évaluant les projections de deux triangles.

IV.4.3.1 Décomposition en triangles

Avant d'étudier la décomposition d'un polygone en triangles, nous définissons la notion de sommet « extérieur », illustrée par la figure IV.8 (a).

Définition 1 Le sommet v_1 est dit *extérieur à $\mathbf{v}$* si et seulement si $\det[\vec{v}_2 - \vec{v}_1, \vec{v}_n - \vec{v}_1] \geq 0$.

Le fait que v_1 soit extérieur permet en général de définir le polygone $\mathbf{v} - \mathbf{t}_1$ et d'écrire $\mathbf{v} = \mathbf{t}_1 \cup (\mathbf{v} - \mathbf{t}_1)$, où $\mathbf{t}_1$ représente le triangle $\{v_1, v_2, v_n\}$. Néanmoins, il existe des situations où v_1 est extérieur à $\mathbf{v}$, mais $\mathbf{t}_1$ n'est pas inclus dans $\mathbf{v}$ (figure IV.8 (c)). Dans ces cas, nous pouvons tout de même définir des ensembles finis de polygones (indexés par I et J , où $I \cap J = \emptyset$) $\mathbf{v}_i$ et $\mathbf{v}_j$ tels que :

- (i) $\forall k \in I \cup J$, v_1 n'est pas un sommet de $\mathbf{v}_k$,
- (ii) $\forall k, l \in I \cup J$, $k \neq l$, $\text{Int}(\mathbf{v}_k \cap \mathbf{v}_l) = \emptyset$,
- (iii) $\mathbf{v} = \cup_{i \in I} \{\mathbf{v}_i\} - \cup_{j \in J} \{\mathbf{v}_j\} + \mathbf{t}_1$.

De même, si v_1 est intérieur à $\mathbf{v}$, nous pouvons décomposer $\mathbf{v}$ sous la forme $\cup_{i \in I} \{\mathbf{v}_i\} - \cup_{j \in J} \{\mathbf{v}_j\} - \mathbf{t}_1$, où le sommet v_1 n'appartient qu'au triangle $\mathbf{t}_1$.

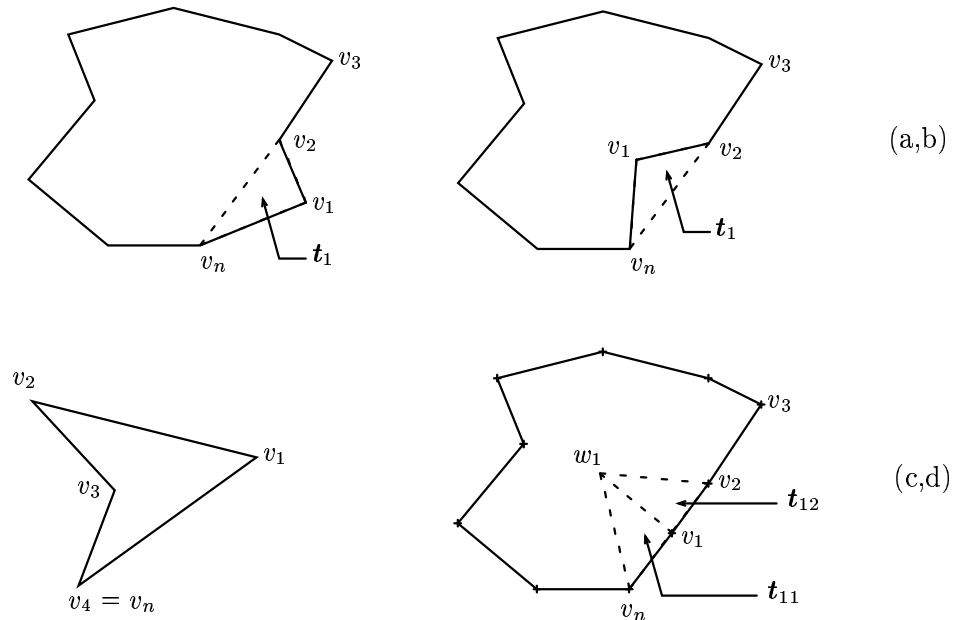


Figure IV.8 – Statut du sommet v_1 de $\mathbf{v}$. a) et b) Sommet extérieur/intérieur à $\mathbf{v}$, c) Sommet extérieur qui ne vérifie pas la propriété $\mathbf{v} = \mathbf{t}_1 \cup (\mathbf{v} - \mathbf{t}_1)$, d) Sommet singulier : $\mathbf{t}_1$ est vide.

IV.4.3.2 Principe de la mise à jour

Considérons à présent une projection de $\mathbf{v}$. D'après les développements précédents,

$$\forall t \in \mathbb{R}, p_{\mathbf{v}}(t) = \varepsilon_1 p_{\mathbf{t}_1}(t) + \sum_{i \in I} p_{\mathbf{v}_i}(t) - \sum_{j \in J} p_{\mathbf{v}_j}(t), \quad (\text{IV.35})$$

où $\varepsilon_1 = \text{sgn}(\det[\vec{v}_2 - \vec{v}_1, \vec{v}_n - \vec{v}_1])$ indique si v_1 est extérieur ou intérieur à $\mathbf{v}$. Par conséquent, si l'on déplace v_1 en v'_1 , la remise à jour de $p_{\mathbf{v}}(t)$ consiste à

- Déterminer ε'_1 , le caractère intérieur ou extérieur de v'_1 .
- Former le triangle $\mathbf{t}'_1 = (v_n, v'_1, v_2)$ et calculer analytiquement sa projection.
- Mettre à jour la projection $p_{\mathbf{v}'}(t) = p_{\mathbf{v}}(t) + \varepsilon'_1 p_{\mathbf{t}'_1}(t) - \varepsilon_1 p_{\mathbf{t}_1}(t)$.

Bien sûr, tout cela reste valable dans le cas singulier illustré sur la figure IV.8 (d), où v_1 et ses voisins sont alignés. Dans ce cas, le triangle $\mathbf{t}_1$ est aplati et sa projection est nulle.

Nous obtenons finalement une procédure de remise à jour très simple, car elle consiste à projeter deux triangles. De plus, cette opération est très peu coûteuse car analytique (voir l'annexe C pour le détail les calculs des projections d'un triangle).

IV.4.3.3 Déplacements infinitésimaux

Ce paragraphe étudie un cas particulier du précédent : nous considérons un déplacement infinitésimal du v_1 pour traiter l'existence et le calcul de $\partial p_{\mathbf{v}}(t)/\partial v_1$. Les résultats donnés ci-dessous sont notamment utilisés au § IV.4.4 pour l'étude de $\nabla \mathcal{J}$.

Dans le cas où v_1 est un sommet régulier, l'existence de $\partial p_{\mathbf{v}}(t)/\partial v_1$ est équivalente à celle de $\partial p_{\mathbf{t}_1}(t)/\partial v_1$, et nous pouvons écrire $\partial p_{\mathbf{v}}(t)/\partial v_1 = \varepsilon_1 \partial p_{\mathbf{t}_1}(t)/\partial v_1$ car un déplacement infinitésimal de v_1 ne modifie pas son caractère extérieur ou intérieur.

En revanche, le cas singulier impose de passer par une autre décomposition de $\mathbf{v}$, car la notion de sommet extérieur est mal définie. À cet effet, nous introduisons un point fictif w_1 , placé sur la demi-droite d'extrémité v_1 et directement orthogonale à $\vec{v}_1 - \vec{v}_n$. En notant $\mathbf{t}_{11} = \{v_n, v_1, w_1\}$ et $\mathbf{t}_{12} = \{v_1, v_2, w_1\}$, nous pouvons montrer que $\mathbf{v}$ se décompose sous la forme $\mathbf{v} = \mathbf{t}_{11} + \mathbf{t}_{12} + \cup_{i \in I} \{\mathbf{v}_i\} - \cup_{j \in J} \{\mathbf{v}_j\}$, où les polygones $\mathbf{v}_i$ et $\mathbf{v}_j$ sont constitués de sommets de $\mathbf{v} \cup \{w_1\}$, et vérifient les propriétés (i) et (ii). Ainsi, seules les remises à jour de $p_{\mathbf{t}_{11}}(t)$ et $p_{\mathbf{t}_{12}}(t)$ sont nécessaires à celle de $p_{\mathbf{v}}(t)$, et nous avons $\forall t, \partial p_{\mathbf{v}}(t)/\partial v_1 = \partial p_{\mathbf{t}_{11}}(t)/\partial v_1 + \partial p_{\mathbf{t}_{12}}(t)/\partial v_1$.

IV.4.4 Calcul du gradient du critère

Nous déduisons de ce qui précède une technique originale d'évaluation exacte du gradient de $\mathcal{J}$. En effet, définir les dérivées partielles de $\mathcal{J}$, c'est analyser l'effet d'un déplacement infinitésimal de chaque sommet (noté dv_i pour le i -ème sommet) sur $p_{\mathbf{v}}(t)$, donc sur $\mathcal{J}(\mathbf{v})$.

Cette stratégie n'est pas la seule possible. Battle et al. (1998) préconisent des techniques de différenciation adjointe pour le même type de critère, en ayant recours à une pixellisation de la scène. Ces techniques sont efficaces et fournissent un résultat exact si le calcul des projections est exact, mais nous préférons dériver un calcul direct, car il se déduit naturellement de la mise à jour des projections. Nous donnons le principe de l'évaluation de $\nabla \mathcal{J} = \nabla \mathcal{K} + \lambda \nabla \mathcal{R}$, et nous renvoyons le lecteur à l'annexe C pour les éléments calculatoires. $\nabla \mathcal{J}$ est défini sous sa forme complexe par :

$$\nabla \mathcal{J}(\mathbf{v}) = (\partial \mathcal{J}/\partial v_1, \dots, \partial \mathcal{J}/\partial v_n)^t \in \mathbb{C}^n, \quad (\text{IV.36})$$

où $\partial \mathcal{J}/\partial v_j = \partial \mathcal{J}/\partial x_j + i \partial \mathcal{J}/\partial y_j$. Le cas du terme de régularisation $\mathcal{R}(\mathbf{v})$ est le plus simple, car il dépend directement des sommets v_j (voir § C.2). Dans ce qui suit, nous étudions particulièrement le gradient du terme d'attache aux données. Nous écrivons ce terme sous la forme

$$\mathcal{K}(\mathbf{v}) = \|\mathbf{d} - \mathcal{A}(\mathbf{v})\|^2 = \sum_{k=1}^m \sum_{j=1}^r [d_j^k - \mathcal{A}_j^k(\mathbf{v})]^2, \quad (\text{IV.37})$$

où les termes d_j^k sont les données et $\{\mathcal{A}_j^k(\mathbf{v})\}$ désigne les projections du polygone $\mathbf{v}$, c'est-à-dire les valeurs échantillonnées des signaux $p_{\theta_k}(t)$. L'équation ci-dessus fait intervenir tous les sommets de $\mathbf{v}$ de façon analogue. Par conséquent, l'existence et le calcul de $\nabla \mathcal{K}$ se déduisent de l'étude de la quantité

$$\partial p_{\mathbf{v}}(t)/\partial v_1, \quad (\text{IV.38})$$

pour une direction de projection θ arbitraire. Or, nous avons vu au § IV.4.3 que ce problème est équivalent à celui de l'existence et du calcul de $\partial p_{\mathbf{t}_1}(t)/\partial v_1$, sauf dans les cas singuliers. Nous énonçons finalement la proposition suivante, dont la preuve est apportée dans l'annexe C.

Proposition 5 *Pour une projection fixée, la quantité (IV.38) est définie si et seulement si $t \neq \langle \vec{v}_1, \mathbf{w} \rangle$. En $t = \langle \vec{v}_1, \mathbf{w} \rangle$, les dérivées partielles à droite et à gauche sont toutefois définies.*

$\mathcal{K}(\mathbf{v})$ est donc différentiable par rapport à v_1 si $\forall k \in \{1, \dots, m\}, \forall j \in \{1, \dots, r\}, t_k + (j-1)\delta t_k \neq \langle \vec{v}_1, \mathbf{w}_{\theta_k} \rangle$, et nous avons dans ce cas :

$$\partial \mathcal{K}(\mathbf{v})/\partial v_1 = 2 \sum_{k=1}^m \sum_{j=1}^r \mathcal{A}_j^k(\mathbf{v}) \partial \mathcal{A}_j^k/\partial v_1 - 2 \sum_{k=1}^m \sum_{j=1}^r d_j^k \partial \mathcal{A}_j^k/\partial v_1. \quad (\text{IV.39})$$

Le premier terme de cette expression fait intervenir les projections $\mathcal{A}_j^k(\mathbf{v})$ du polygone $\mathbf{v}$, que nous pouvons évaluer de façon exacte, mais non analytique en fonction de $\mathbf{v}$ (voir § IV.4.2). Par conséquent, nous ne pouvons dériver une expression analytique de $\partial \mathcal{K}(\mathbf{v})/\partial v_1$ lorsque cette quantité est définie.

Conclusions. Nous avons développé une procédure de calcul exact (bien que non analytique) du gradient de $\mathcal{J}$. De plus, cette évaluation est très simple ; en supposant la projection de $\mathbf{v}$ calculée, il suffit de calculer les dérivées partielles des projections de triangles par rapport à leurs sommets, dont l'expression est analytique. Dans l'optique de l'optimisation de $\mathcal{J}$, nous avons exprimé une condition nécessaire et suffisante simple pour l'existence de $\nabla \mathcal{J}(\mathbf{v})$ (proposition 5) : $\nabla \mathcal{J}(\mathbf{v})$ est défini si aucun des projetés des sommets de $\mathbf{v}$ ne coïncide avec le centre d'un pixel détecteur. Il y a donc une infinité de configurations $\mathbf{v}$ en lesquelles $\mathcal{J}$ n'est pas différentiable par rapport aux coordonnées d'un de ses sommets. Néanmoins, l'ensemble de ces configurations est de mesure nulle, et pour l'une d'elles, il existe des variables par rapport auxquelles $\mathcal{J}$ possède une dérivée partielle. En d'autres termes, si l'un des projetés du sommet v_i coïncide avec le centre d'un pixel détecteur, $\mathcal{J}$ n'est pas dérivable par rapport à la x_j, y_j et v_j pour $j = i$ ou j voisin de i , mais est en général dérivable par rapport aux coordonnées des autres sommets.

IV.5 Conclusions

Dans ce chapitre, nous avons développé la reconstruction d'objets polygonaux à grand nombre de sommets à partir de données tomographiques. Nous avons opté pour l'estimation bayésienne des positions des sommets, en incorporant de l'information *a priori* sur la douceur locale du contour. Les difficultés que pose cette estimation sont de plusieurs ordres. D'une part, l'estimation des sommets au sens du MAP pose le problème de la minimisation d'une fonction généralement multimodale. D'autre part, la déformation de polygones nécessite la vérification de la validité de ces formes, au sens où elles ne doivent pas s'intersecter. C'est une difficulté qui n'est pas inhérente au modèle polygonal, les modèles splines en général, et bien d'autres modèles géométriques déformables sont soumis à ce type de problèmes (Senasli et al., 2000). L'originalité de notre contribution concerne en partie ce dernier point, généralement abordé *via* une approximation de la scène polygonale par un ensemble de pixels (Hanson et al., 1997). Le traitement des auto-intersections est sous-jacent à l'algorithme d'optimisation que nous utilisons.

Pour limiter la charge calculatoire, nous optons pour un algorithme itératif d'optimisation locale. Dans cette classe d'algorithmes, nous avons mis en valeur un algorithme par blocs (où les blocs sont les sommets du polygone), qui permet de gérer la « validité » des polygones. Compte tenu du caractère non convexe de l'énergie *a posteriori*, l'initialisation de l'algorithme est un point clé, qui conditionne la reconstruction finale. Cette initialisation est fournie par une méthode plus simple, développée au chapitre III, qui estime au sens des moindres carrés (contraints) les paramètres d'un modèle global harmonique.

Dans le chapitre V, nous étendons le modèle polygonal au cas 3D, en proposant une structure similaire pour l'estimation des sommets d'un polyèdre. Nous montrons l'intérêt de cette extension, en étudiant les difficultés à la fois théoriques et algorithmiques qu'elle impose.

IV.A Caractère non convexe de $\mathcal{K}$

Nous montrons sur un exemple simple que les opérateurs $\mathcal{A}$ et $\mathcal{K}$ ne possèdent pas toujours des propriétés favorables. Pour cela, nous considérons une seule projection $p_0(t)$ suivant l'angle $\theta = 0$, *i.e.*, suivant la direction verticale, d'un triangle (v_1, v_2, v_3) dont le côté (v_1, v_2) est parallèle à l'axe $(0, x)$, comme illustré en figure IV.9 (a). Nous montrons en particulier que :

- $\mathcal{A}$ et $\mathcal{K}$ ne sont pas continues en certaines valeurs de $\mathbf{v}$.
- $\mathcal{A}(\mathbf{v})$ n'est pas affine.
- $\mathcal{K}(\mathbf{v})$ est non convexe en général, voire multimodal.

Comme nous ne considérons qu'une projection, les opérateurs $\mathcal{A}$ et $\mathcal{K}$ s'expriment par

$$\mathcal{A}(\mathbf{v}) = [p_0(t_1), \dots, p_0(t_r)]^t \text{ et } \mathcal{K}(\mathbf{v}) = \sum_{i=1}^r [d_i - p_0(t_i)]^2, \quad (\text{IV.40})$$

où les t_i représentent les positions des détecteurs. Supposons que $x_1 < x_3 < x_2$, $p_0(t)$ s'exprime alors par morceaux :

$$p_0(t) = \begin{cases} 0 & \text{si } t \leq x_1 \text{ ou } t \geq x_2, \\ |y_3 - y_1| (t - x_1) / (x_3 - x_1) & \text{si } x_1 < t \leq x_3, \\ |y_3 - y_1| (x_2 - t) / (x_2 - x_3) & \text{si } x_3 < t < x_2. \end{cases} \quad (\text{IV.41})$$

IV.A.1 Cas $r = 1$

Plaçons nous d'abord dans le cas où $r = 1$, fixons la position de v_1, v_2 et y_3 comme indiqué sur la figure IV.9 (a). Si nous prenons $t_1 = x_2$, alors $\mathcal{A}(\mathbf{v}) = p_0(t_1)$ est discontinue en tant que fonction de x_3 . En effet, $\mathcal{A}(\mathbf{v}) = 0$ pour $x_3 < t_1$, et $\mathcal{A}(\mathbf{v}) = y_3 - y_1$ pour $x_3 = t_1$. Par conséquent, $\mathcal{A}(\mathbf{v})$ et $\mathcal{K}(\mathbf{v})$ ne sont pas continues sur $\mathbb{A}_3$.

Prenons à présent le cas où $t_1 \in]x_1, x_2[$. Les caractères non affine de $\mathcal{A}(\mathbf{v})$ et non convexe de $\mathcal{K}(\mathbf{v})$ sont alors très nets à partir de la formulation (IV.41). En effet, à t_1, v_1 et v_2 fixés, la dépendance de $\mathcal{A}(\mathbf{v}) = p_0(t_1)$ par rapport à y_3 est affine par morceaux, mais la dépendance par rapport à x_3 est hyperbolique par morceaux, donc non convexe. En notant $x_3 \rightarrow a(x_3) = \mathcal{A}(\mathbf{v})$ une telle fonction, nous avons d'après (IV.41) :

$$a(x_3) = \begin{cases} |y_3 - y_1| (x_2 - t_1)/(x_2 - x_3) & \text{si } x_3 < t_1, \\ |y_3 - y_1| (t_1 - x_1)/(x_3 - x_1) & \text{sinon.} \end{cases} \quad (\text{IV.42})$$

Nous illustrons cette fonction sur la figure IV.9 (b).

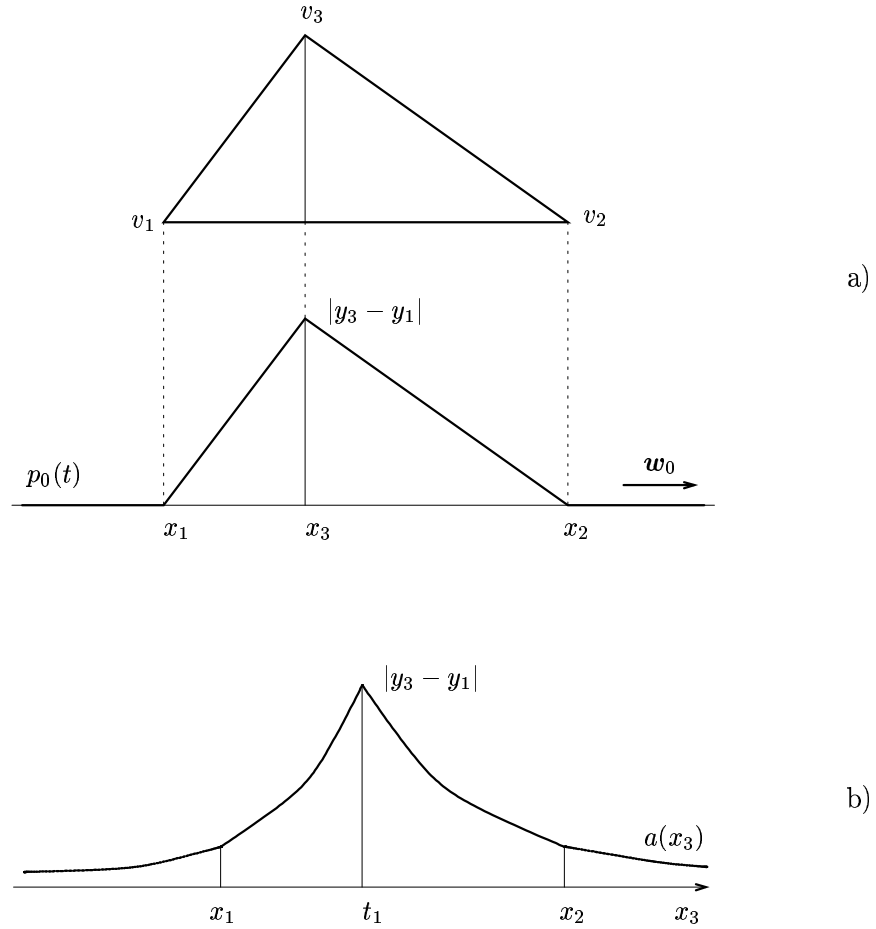


Figure IV.9 – Structure des projections d'un triangle. a) Projection d'un triangle (v_1, v_2, v_3) suivant la direction $\theta = 0$. b) Représentation de la fonction $a(x_3)$, *i.e.*, de l'évolution de $p_0(t_1)$ en fonction de x_3 , pour t_1 fixé dans (x_1, x_2) .

IV.A.2 Cas $r > 1$

Nous avons montré ci-dessus que $\mathcal{K}(\mathbf{v})$ est non convexe sur un exemple simple (pour fixer les idées, si $d_1 = 0$, $\mathcal{K}(\mathbf{v}) = a^2(x_3)$, et $x_3 \rightarrow \mathcal{K}(\mathbf{v})$ est non convexe, car est du même type que $a(x_3)$, représentée sur la figure IV.9 (b)). En outre, cette fonction $\mathcal{K}(\mathbf{v})$ possède plusieurs modes à l'infini. Il est très simple de définir des configurations à 2 ou 3 données pour lesquelles $\mathcal{K}(\mathbf{v})$ possède plusieurs modes finis. En effet, la somme de plusieurs fonctions du type $p_0(t_i)^2$, d'allure similaire à la courbe IV.9 (b), mène souvent à des fonctions à plusieurs maxima et/ou minima locaux.

CHAPITRE V

APPROCHE POLYÉDRIQUE

- V.1 Introduction
- V.2 Modèle polyédrique
- V.3 Optimisation du critère
- V.4 Étude des projections
- V.5 Extraction de mini-polyèdres
- V.6 Conclusions
- V.A Preuve de la proposition 7

V.1 Introduction

Dans ce chapitre, nous développons spécifiquement le modèle polyédrique pour la reconstruction tomographique de contours fermés. Le point de départ de cette étude est la reconstruction de formes polygonales à partir de données 2D, que nous avons développée au chapitre IV. Les principes des méthodes 2D et 3D sont proches, mais le problème 3D impose de nombreuses adaptations à la fois théoriques et algorithmiques. Pour ne pas alourdir sa description, nous nous basons sur les développements du chapitre IV, en insistant sur les spécificités du modèle 3D.

L'organisation du chapitre est analogue à celle du chapitre IV. Dans un premier temps, nous définissons le modèle polyédrique par un ensemble de sommets et un ensemble de facettes. Par simplicité, nous fixons les facettes, et le problème de reconstruction consiste finalement à estimer les positions des sommets directement à partir des données. Cette estimation est réalisée dans un contexte bayésien, qui permet de prendre en compte des informations *a priori* sur le contour (douceur locale). En outre, nous montrons que le problème se ramène à la minimisation d'un critère régularisé qui porte sur les coordonnées des sommets. Comme dans le cas 2D, les parties algorithmiques constituent l'essentiel de notre contribution par rapport aux travaux existants (Battle et al., 1998). Ainsi, la section V.3 étudie la mise en œuvre de cette optimisation et souligne les difficultés techniques qui en découlent (il s'agit notamment de contrôler que l'objet ne s'intersecte pas). Afin de les résoudre, nous proposons une étude originale et détaillée des projections du modèle (§ V.4). Nous verrons qu'un point essentiel à sa manipulation est l'extraction de formes locales à chaque sommet (appelées mini-polyèdres) et décomposables en une partition de tétraèdres (§ V.5).

Les parties V.4 et V.5 de ce chapitre sont davantage d'ordre algorithmique, mais elles sont essentielles à la manipulation du modèle. Toutefois, les contributions les plus techniques sont développées dans les annexes D et E.

V.2 Modèle polyédrique

Dans un premier temps, nous étudions l'extension 3D de la méthode des moments, présentée au § IV.2.1. Dans le cas 2D, nous avons vu que cette méthode permet de reconstruire des formes peu complexes car l'estimation des moments est très sensible au bruit. Nous concevons donc au § V.2.1 une méthode analogue pour la reconstruction de surfaces 3D *simples*. Par la suite, nous développons une approche originale pour la reconstruction de polyèdres à plus grand nombre de sommets.

V.2.1 Méthode des moments

De même qu'au § IV.2.1, nous procédons en deux étapes. Nous calculons à partir des projections, les moments géométriques de $\mathcal{P}$

$$\mu_{ijk} = \iiint_{\mathcal{P}} x^i y^j z^k dx dy dz, \quad i, j, k \geq 0 \quad (\text{V.1})$$

jusqu'à un certain ordre p (*i.e.*, pour $i + j + k \leq p$), puis nous estimons un contour $\widehat{\mathcal{C}}$ à partir des moments estimés $\widehat{\mu}$. Le premier sous-problème est résolu dans la contribution (Soussen et Mohammad-Djafari, 1999c). Nous montrons en particulier qu'il est possible de reconstruire les moments géométriques de $\mathcal{P}$ aux ordres $0, \dots, p$ à partir de $p+1$ projections, sauf dans des configurations particulières de projection. Le lecteur intéressé par ces aspects pourra consulter l'annexe B, et plus précisément les sections B.3 et B.3.4. En pratique, nous prôtons une méthode des moindres carrés pour l'estimation des moments μ_{ijk} à partir des moments 2D des données.

La résolution théorique du second sous-problème, *i.e.*, la reconstruction d'un contour 3D à partir des moments μ_{ijk} jusqu'à un certain ordre p , est plus complexe. Tout d'abord, considérons le problème direct, qui calcule les moments de l'objet défini par un contour fermé. Très peu de formes paramétriques mènent à des moments géométriques qui s'expriment de façon simple en fonction des paramètres. Pour les polyèdres, Timmer et Stern (1980) et Mirtich (1996) évaluent de façon exacte les moments à partir des sommets. En revanche, l'inversion de telles relations ne semble pas envisageable, car ces expressions sont complexes, non linéaires en fonction des coordonnées des sommets, et dépendent aussi des facettes du polyèdre. Nous optons finalement pour la reconstruction d'*ellipsoïdes*, qui sont des formes simples entièrement déterminées par leurs moments jusqu'à l'ordre 2 (voir § III.4.1). Par ailleurs, l'estimation des moments d'ordre supérieur à partir de données bruitées est très sensible au bruit, donc peu informative. Par conséquent, nous n'envisageons pas la reconstruction de formes plus complexes à partir de moments d'ordres élevés.

V.2.2 Modèle proposé

Contrairement au cas bidimensionnel, une séquence ordonnée de sommets $V_i \in \mathbb{R}^3$ ne définit pas un polyèdre. En effet, il est indispensable d'y adjoindre une description des facettes. Un polyèdre $\mathcal{P}$ à n sommets et q facettes *triangulaires* est donc spécifié par :

$$\mathcal{P} \equiv \{\mathbf{V}, \mathcal{F}\}, \quad (\text{V.2})$$

où les ensembles de sommets et de facettes sont définis par :

$$\mathbf{V} = [V_1^t, V_2^t, \dots, V_n^t]^t \in \mathbb{R}^{3n}, \quad V_i = (x_i, y_i, z_i)^t, \quad (\text{V.3})$$

$$\mathcal{F} = \{\mathcal{F}_i, i = 1, \dots, q\}, \quad \text{où } \mathcal{F}_i \in \{1, \dots, n\}^3. \quad (\text{V.4})$$

En particulier, nous dénotons généralement l'objet polyédrique par $\mathcal{P}$, et non par $\mathbf{V}$ comme au chapitre IV, pour insister sur le fait que $\mathbf{V}$ ne définit pas le polyèdre à lui tout seul. Cependant, $\mathbf{V}$ interviendra davantage au niveau algorithmique.

Le choix de facettes triangulaires est fait pour des raisons de mise en œuvre, nous y reviendrons au § V.4.1. Les facettes $\mathcal{F}_i$ sont définies de façon à engendrer des normales $\mathbf{f}_i \propto \mathbf{V}_{i1} \mathbf{V}_{i2} \wedge \mathbf{V}_{i1} \mathbf{V}_{i3}$ qui pointent vers l'extérieur¹ (voir la figure V.5 page 91). Nous notons finalement $\mathbb{P}_{nq}$ l'ensemble des polyèdres à n sommets et q facettes triangulaires, puis nous définissons le sous-ensemble $\mathbb{A}_{nq}$ des polyèdres *acceptables* (ou sans auto-intersection) par :

¹Notons qu'à facettes $\mathcal{F}_i$ fixées, les normales $\mathbf{f}_i$ pointent toujours vers l'extérieur quels que soient les déplacements ultérieurs des sommets de $\mathbf{V}$, pourvu que cette orientation soit correcte avec la configuration initiale $\mathbf{V}^{(0)}$.

$$\mathbb{A}_{nq} = \left\{ \mathcal{P} = \{\mathbf{V}, \mathcal{F}\} \in \mathbb{P}_{nq}, \forall (i, j) \in \{1, \dots, q\}, i \neq j, i \not\sim j, \mathcal{F}_i \cap \mathcal{F}_j = \emptyset \right\}. \quad (\text{V.5})$$

Pour la segmentation d'images 3D modélisées par des maillages, des conditions plus strictes que (V.5) ont été conçues. Par exemple, Lachaud et Montanvert (1998) associent un paramètre de régularité δ à un maillage, via les conditions suivantes :

$$\forall i, j \in \{1, \dots, n\}, \begin{cases} i \sim j \Rightarrow \delta < \|\mathbf{V}_i \mathbf{V}_j\| < 2.5 \delta, \\ \neg(i \sim j) \Rightarrow 2.5 \delta / \sqrt{3} < \|\mathbf{V}_i \mathbf{V}_j\|. \end{cases} \quad (\text{V.6})$$

Ces conditions imposent notamment une distance minimale entre deux sommets du polyèdre. Elles génèrent donc des objets plus réguliers, et qui vérifient bien sûr (V.5). Nous n'utilisons pas un tel modèle car nous ne connaissons pas de procédure rapide qui permette de vérifier (V.6).

Pour compléter la définition du modèle, nous définissons les projections de $\mathcal{P}$ par $\mathbf{p} = \mathcal{A}(\mathcal{P})$. Ces projections sont calculées en les pixels détecteurs considérés comme ponctuels (chaque pixel est assimilé à son centre) comme nous l'avons vu au chapitre I. Dans les cas où les facettes sont fixées, l'opérateur $\mathcal{A}$ agit uniquement sur les positions des sommets, et nous écrivons par abus de notations $\mathbb{A}_n \equiv \mathbb{A}_{nq}$ et

$$\mathcal{A} : \mathbb{A}_n \longrightarrow \mathbb{R}^M \quad (\text{V.7})$$

$$\mathbf{V} \longrightarrow \mathcal{A}(\mathbf{V}) = \mathcal{A}(V_1, \dots, V_n), \quad (\text{V.8})$$

où M est le nombre total de pixels détecteurs. De même, le terme d'adéquation aux données s'écrit

$$\mathcal{K}(\mathbf{V}) = \mathcal{K}(\mathcal{P}) = \|\mathbf{d} - \mathcal{A}(\mathbf{V})\|^2. \quad (\text{V.9})$$

V.2.3 Estimation des sommets au sens du MAP

Principe. En l'absence d'information *a priori* particulière sur la forme de l'objet, nous choisissons de favoriser sa *douceur locale*. De même que dans le cas 2D, nous préconisons une énergie $\mathcal{R}(\mathbf{V})^2$ à caractère markovien, qui pénalise les fortes courbures de $\mathcal{P}$. De tels modèles de régularisation ont été développés abondamment dans le cas de maillages 3D pour la segmentation d'images (Bulpitt et Efford, 1996; Delingette, 1997), et seulement récemment pour la reconstruction tomographique (Battle et al., 1998). De même que dans le cas 2D, $\mathcal{R}$ prend la forme

$$\mathcal{R}(\mathbf{V}) = \sum_{i=1}^n \phi(\kappa_i(\mathbf{V})), \quad (\text{V.10})$$

où les termes $\kappa_i(\mathbf{V})$ sont les courbures locales approchées en les sommets V_i . La fonction ϕ retenue est toujours la quadratique $\phi(t) = t^2$.

Définition des courbures. La courbure en V_i peut être définie de plusieurs façons. Bulpitt et Efford (1996) proposent la distance de V_i au plan moyen de ses voisins, normalisée par la taille du voisinage. Nous ne retenons pas cette mesure car elle est approximative (la taille du voisinage est un paramètre arbitraire), et fait intervenir des calculs intermédiaires de normales. Nous optons pour le calcul exact de l'*angle solide* $A_i(\mathbf{V})$ en V_i , qui généralise celui de l'angle en un sommet d'un polygone.

²En toute rigueur, nous devrions écrire $\mathcal{R}(\mathcal{P})$, car cette fonctionnelle dépend explicitement des relations d'adjacence entre les sommets de $\mathcal{P}$.

Définition 2 (*Liu et Joe, 1994*) L'angle solide $A_i(\mathbf{V})$ est l'aire de la surface obtenue en projetant tous les sommets voisins de V_i sur la sphère unité centrée en V_i .

Cette définition est illustrée sur la figure V.1 dans le cas d'un tétraèdre. La quantité $A_i(\mathbf{V})$ est une aire, elle prend donc des valeurs dans $(0, 4\pi)$. Une valeur proche de 0 ou 4π correspond à un sommet singulier de $\mathcal{P}$, et une valeur de 2π correspond à une partie plane de la surface de $\mathcal{P}$. Enfin, V_i est intérieur à $\mathcal{P}$ si et seulement si $A_i(\mathbf{V})$ est inférieur à 2π . Nous définissons finalement la courbure non signée en V_i par

$$\kappa_i(\mathbf{V}) = \kappa(A_i(\mathbf{V})) = 1 + \cos(A_i(\mathbf{V})/2). \quad (\text{V.11})$$

La fonction $\kappa(A)$, définie sur $(0, 4\pi)$, est à valeurs dans $(0, 2)$ et décroissante sur $(0, 2\pi)$. De plus, elle est symétrique par rapport à 2π ($\kappa(A) = \kappa(4\pi - A)$), si bien que deux angles solides supplémentaires (de somme 4π) sont logiquement pénalisés de la même façon. Ils ont physiquement la même courbure (la seule différence est que l'un des sommets est intérieur à $\mathcal{P}$ et l'autre est extérieur).

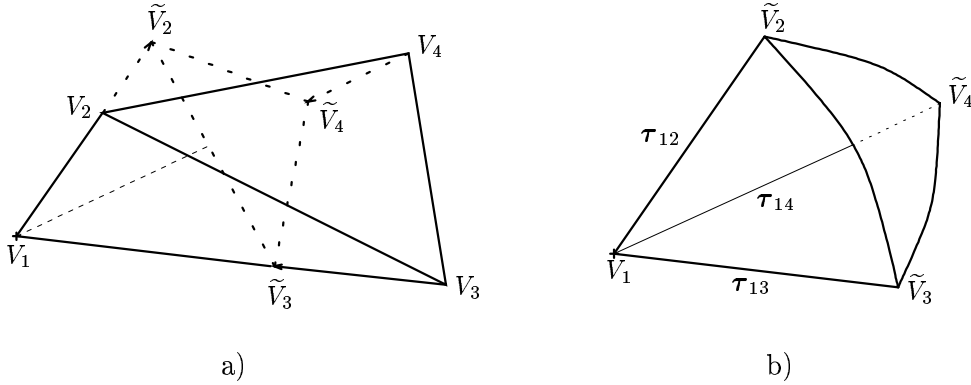


Figure V.1 – Angle solide d'un tétraèdre $\mathcal{T}$ en son sommet V_1 . a) Transformation de $\mathcal{T}$ en $\tilde{\mathcal{T}}$, le tétraèdre de sommets V_1 et $\tilde{V}_2, \tilde{V}_3, \tilde{V}_4$. Les sommets $\tilde{V}_i$ sont les projections de V_i sur la sphère unité centrée en V_1 . b) $A_1(\mathcal{T})$ est défini par l'aire du triangle sphérique $(\tilde{V}_2, \tilde{V}_3, \tilde{V}_4)$.

Calcul des angles solides. L'angle solide $A_i(\mathbf{V})$ est directement lié au cône $\Gamma_i(\mathcal{P})$ correspondant. Ce cône s'appuie sur les demi-droites $[V_i, V_{i \rightarrow 1}), \dots, [V_i, V_{i \rightarrow n_i})$ qui supportent les arêtes de $\mathcal{P}$ adjacentes à V_i . En notant $\boldsymbol{\tau}_{ij}$ les vecteurs directeurs de ces demi-droites ($\boldsymbol{\tau}_{ij} = \mathbf{V}_i \mathbf{V}_{i \rightarrow j} / \|\mathbf{V}_i \mathbf{V}_{i \rightarrow j}\|$), et en classant les voisins de V_i par ordre d'adjacence ($V_{i \rightarrow j}$ et $V_{i \rightarrow j+1}$ sont voisins dans $\mathcal{P}$), nous déduisons une partition de $\Gamma_i(\mathcal{P})$ en un ensemble de cônes élémentaires :

$$\Gamma_i(\mathcal{P}) = \bigcup_{j=1}^{n_i} \Gamma_{ij}(\mathcal{P}), \quad (\text{V.12})$$

où chaque cône $\Gamma_{ij}(\mathcal{P})$ s'appuie sur V_i et les trois directions $(\boldsymbol{\tau}_{ij}, \boldsymbol{\tau}_{i,j+1}, \boldsymbol{\tau}_i)$, et $\boldsymbol{\tau}_i$ désigne une direction fictive commune aux cônes Γ_{ij} (voir l'annexe E). En vertu de la décomposition (V.12), $A_i(\mathbf{V})$ est égal à la somme des angles solides des cônes $\Gamma_{ij}(\mathcal{P})$, *i.e.*, la somme des angles solides en V_i des tétraèdres associés :

$$A_i(\mathbf{V}) = \sum_{j=1}^{n_i} A_i(V_i, V_{i \rightarrow j}, V_{i \rightarrow j+1}, W_i), \quad (\text{V.13})$$

où W_i désigne un point quelconque de la droite orientée $(V_i, \boldsymbol{\tau}_i)$. Nous montrons dans l'annexe E que l'évaluation des termes (V.13) est directe.

Critère global. Finalement, le critère global s'écrit $\mathcal{J}(\mathbf{V}) = \mathcal{K}(\mathbf{V}) + \lambda\mathcal{R}(\mathbf{V})$, où le terme de régularisation s'exprime en fonction des angles solides en les sommets V_i :

$$\mathcal{R}(\mathbf{V}) = \sum_{i=1}^n \phi(1 + \cos(A_i(\mathbf{V})/2)), \quad (\text{V.14})$$

et $\phi(t) = t^2$. Les angles solides $A_i(\mathbf{V})$ s'expriment en fonction des angles solides des tétraèdres élémentaires $(V_i, V_{i \rightarrow j}, V_{i \rightarrow j+1}, W_i)$ par (V.13), et ces derniers sont évalués en annexe E (voir l'expression (E.5) pour la tangente de l'angle solide moitié d'un tétraèdre). Comme dans le cas 2D, l'estimateur choisi est le MAP, noté $\hat{\mathbf{V}}_{\text{MAP}}$ ou $\hat{\mathbf{V}}$, et minimise $\mathcal{J}(\mathbf{V})$ sur $\mathbb{A}_n$.

V.3 Optimisation du critère

Nous avons ramené ci-dessus la reconstruction du polyèdre $\mathcal{P} = \{\mathbf{V}, \mathcal{F}\}$ à la minimisation contrainte (sur $\mathbb{A}_n$) de l'énergie *a posteriori* $\mathcal{J}(V_1, \dots, V_n)$. Comme dans le cas 2D, ce critère est généralement non convexe, et nous envisageons une stratégie itérative déterministe pour son optimisation locale. Avant d'aborder les éléments techniques de l'optimisation, nous sélectionnons un algorithme spécifique du premier ordre.

V.3.1 Techniques locales d'optimisation

Nous utilisons les techniques déjà décrites au § IV.3. Il s'agit d'algorithmes de descente sur $\mathbf{V}$, et de descente par blocs (de $\mathbb{R}^3$), où chaque bloc représente un sommet V_i du polyèdre. Ces algorithmes nécessitent la connaissance du gradient de $\mathcal{J}$, nous développons donc au § V.4 (et dans l'annexe D) une méthode originale d'évaluation de $\nabla \mathcal{J}$.

Sans revenir sur la description des algorithmes, nous donnons des arguments liés au contexte 3D du problème pour le choix de l'un ou de l'autre. Comme dans le cas 2D, l'algorithme par blocs présente l'avantage de gérer les auto-intersections des polyèdres (voir le § V.4.2 et le tableau V.1 page 88). En revanche, la charge calculatoire associée à une itération de cet algorithme (à un parcours de tous les sommets V_i) est bien plus élevée que celle d'une itération de l'algorithme de descente sur $\mathbf{V}$, en dépit de la simplicité de la mise à jour des projections suite au déplacement d'une *seul* sommet (voir § V.4.2). Cela s'explique par la proposition suivante.

Proposition 6 *Une itération de l'algorithme de descente sur $\mathbf{V}$, i.e., une étape de minimisation de ligne (IV.22) nécessite d'évaluer deux projections de polyèdres.*

Preuve 6 À l'itération k de cet algorithme, nous procédons à une minimisation *approchée* sur $\mathbb{R}_+$ de la fonction $g : s \rightarrow \mathcal{J}(\mathbf{V}^{(k)} - s \nabla \mathcal{J}(\mathbf{V}^{(k)}))$. Nous approchons cette fonction par un polynôme $h(s) = as^3 + bs^2 + cs + d$ de degré 3, dont la minimisation sur $\mathbb{R}_+$ s'effectue analytiquement. Connaissant les quantités $d = g(0) = \mathcal{J}(\mathbf{V}^{(k)})$ et $c = g'(0) = -\|\nabla \mathcal{J}(\mathbf{V}^{(k)})\|^2$, l'approximation de g par h ne nécessite que la détermination de a et b . Pour cela, il suffit de connaître $g(s)$ en deux valeurs positives et distinctes (s_1 et s_2) de s , c'est-à-dire de connaître $\mathcal{J}(\mathbf{V}^{(k)} - s \nabla \mathcal{J}(\mathbf{V}^{(k)}))$ pour $s = s_1, s_2 > 0$. La partie la plus coûteuse de ces calculs est bien sûr l'évaluation des projections de ces polyèdres. $\square$

Le choix de l'algorithme est finalement motivé par sa charge calculatoire, les deux algorithmes donnant des résultats de qualité sensiblement égale. Malgré le fait que la stratégie sommet par sommet permette une vérification du caractère acceptable des reconstructions $(\mathcal{P}^{(k)} \in \mathbb{A}_n, \forall k)$, nous optons donc pour la *stratégie globale* lorsque n est élevé ($n \geq 50$).

V.3.2 Difficultés de mise en œuvre

Outre les difficultés inhérentes aux calculs des projections et de leur gradient, le choix de l'algorithme pose le problème de la gestion des *auto-intersections*, *i.e.*, du caractère acceptable de $\mathcal{P}$ (au sens de la définition (V.5) et de la figure IV.1 dans le cas 2D). Ce problème est rencontré de façon plus générale dès lors que l'on déforme des contours 3D. Une façon de détecter les auto-intersections est l'utilisation d'une voxellisation de l'objet, comme le font Battle et al. (1998). Ils répertorient les facettes qui intersectent chaque voxel, et détectent une auto-intersection si deux facettes non adjacentes intersectent le même voxel. Cette stratégie est efficace, bien qu'elle ne garantisse pas en théorie les caractères acceptable et non acceptable des polyèdres. Cependant, elle s'avère coûteuse car elle nécessite une voxellisation fine et de nombreuses recherches d'intersections voxel-facettes.

Pour notre application, nous n'utilisons pas cette stratégie. Comme en 2D, nous tirons partie de l'étude des projections de $\mathcal{P}$ pour contrôler son caractère acceptable. Les tests qui en résultent diffèrent suivant la technique d'optimisation employée. Considérons la k -ème itération de l'algorithme d'optimisation de $\mathcal{J}$, qui estime $\mathcal{P}^{(k)}$ à partir de $\mathcal{P}^{(k-1)}$:

- Pour la stratégie par blocs (reconstructions à faible nombre de sommets), nous déduisons de la mise à jour des projections un test original d'acceptabilité du polyèdre $\mathcal{P}^{(k)}$ du même type que l'algorithme IV.2 décrit page 66. Nous détaillons ce test au § V.4.2 et par le tableau V.1 (page 88).

- Pour l'algorithme du gradient, nous ne disposons pas d'un tel test. Par contre, nous notons comme dans le cas 2D, que l'algorithme de calcul de $\nabla \mathcal{J}(\mathbf{V}^{(k)})$ ne s'étend pas au cas où $\mathbf{V}^{(k)} \notin \mathbb{A}_n$. Par conséquent, un test d'acceptabilité (condition nécessaire) de $\mathbf{V}^{(k)}$ consiste simplement à comparer les dérivées partielles $\partial \mathcal{J}(\mathbf{V}^{(k)}) / \partial V_i$ à leurs approximations par différences finies.

V.3.3 Conclusions

Nous avons étudié deux stratégies déterministes pour l'optimisation locale du critère. L'une modifie simultanément tous les sommets du polyèdre, alors que l'autre, dite par blocs, estime successivement chacun des sommets. Dans le cas où le polyèdre possède un grand nombre de sommets, nous optons pour la première (technique de descente sur $\mathbf{V}$), nettement moins coûteuse que l'algorithme par blocs. Nous revenons sur ce point au chapitre § VI, où nous quantifions le coût numérique de ces procédures.

Ces deux stratégies sont locales, elles mènent à des solutions très sensibles à l'initialisation. Nous illustrerons ce propos à l'aide de simulations numériques au chapitre VI, en montrant et quantifiant la qualité des reconstructions obtenues sur un même jeu de données à partir d'initialisations différentes. Ces résultats nous mènent finalement à des initialisations de type surfaces harmoniques, reconstruites par la méthode présentée au chapitre III (moindres carrés approchés).

Dans ce qui suit, nous développons les éléments nécessaires à la mise en œuvre des algorithmes décrits ci-dessus. Il s'agit en particulier de l'étude des projections d'un polyèdre, et des problèmes liés à la manipulation de ces objets (manipulation de mini-polyèdres). Comme dans le cas 2D, les parties les plus techniques sont développées en annexe (cf. annexe D).

V.4 Étude des projections

Dans cette partie, nous considérons l'évaluation directe des projections $p_{\theta}(\mathbf{t})$ (où $\mathbf{t}$ est une position fixée du détecteur), et leur mise à jour après déplacements des sommets de $\mathcal{P}$. Ce dernier point conduit à l'évaluation du gradient de $\mathbf{V} \rightarrow p_{\theta}(\mathbf{t})$ (à $\mathbf{t}$ fixé), puis à celui de $\mathcal{J}(\mathbf{V})$. Ces questions sont essentielles car elles interviennent constamment dans les algorithmes utilisés pour la reconstruction.

Pour le calcul direct de $p_{\theta}(\mathbf{t})$, une procédure approchée (déjà développée au § IV.4.1 pour le cas 2D) est envisageable. Elle s'appuie sur la discrétisation de la scène en un ensemble de voxels $\mathbf{f}$ (Battle et al., 1998). Elle s'avère toutefois très technique pour ce qui concerne l'évaluation des valeurs des voxels frontières. Dans ce qui suit, nous lui préférons un calcul exact (et rapide), basé sur les recherches d'intersections rayons-facettes.

V.4.1 Calcul direct

Ce calcul est tout à fait similaire à celui des projections bidimensionnelles d'un polygone, développé au § IV.4.2. Nous rappelons brièvement son principe. Soit $\mathcal{P}$ un polyèdre et $\mathcal{L} = \mathcal{L}_{\mathbf{t},\theta}$ le rayon de projection passant par la source S , orienté par $\mathbf{w} = \mathbf{w}_{\theta}$ et arrivant en le pixel détecteur $\mathbf{t}$. Dans le cas général, $p_{\theta}(\mathbf{t})$ dépend des intersections entrantes et sortantes entre $\mathcal{L}$ et les facettes de $\mathcal{P}$. En omettant le cas singulier où $\mathcal{L}$ est inclus dans le plan support d'une facette (cas traité au § IV.4.2), le nombre d'intersections $I_j = \mathcal{L} \cap \mathcal{F}_j$ est pair. La quantité $p_{\theta}(\mathbf{t})$ est alors égale à la somme des distances $d(I_{2j-1}, I_{2j})$ entre les intersections entrantes et sortantes selon la formule (IV.28).

V.4.1.1 Algorithmes de calcul

De même que dans le cas 2D, nous distinguons les polyèdres convexes des polyèdres quelconques, car des algorithmes rapides existent pour les premiers.

Cas convexe. Pour un polyèdre convexe, il y a soit aucune intersection $\mathcal{L} \cap \mathcal{F}_j$ ($\mathcal{L}$ est extérieur à $\mathcal{P}$), soit exactement deux intersections. Cette remarque mène à une technique analogue à l'algorithme IV.3 (page 71), et qui ne fait intervenir que des intersections rayon-plan (entre $\mathcal{L}$ et $\text{Supp}(\mathcal{F}_j)$).

Cas non convexe. Par contre, le cas non convexe nécessite les évaluations explicites des intersections rayons-facettes. C'est ce facteur qui a motivé le choix de facettes *triangulaires*. L'algorithme d'évaluation est décrit par le tableau IV.4, page 72, et nous explicitons dans le paragraphe suivant l'évaluation de $I_j(l_k) = \mathcal{L} \cap \mathcal{F}_j$.

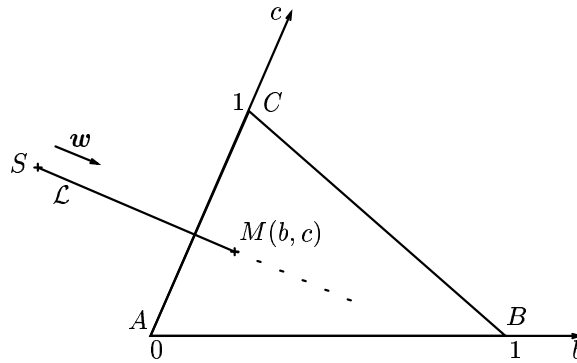


Figure V.2 – Intersection du rayon $\mathcal{L}$ et d'une facette triangulaire $\mathcal{F} = (A, B, C)$.

V.4.1.2 Intersections rayon-triangle

Les recherches d'intersections rayon-triangle sont effectuées comme suit. Prenons une facette triangulaire $\mathcal{F} = (A, B, C)$. Pour déterminer $\mathcal{L} \cap \mathcal{F}$, nous recherchons dans un premier temps l'intersection $\mathcal{L} \cap \text{Supp}(\mathcal{F})$. Cette intersection est soit vide, soit confondue à $\mathcal{L}$, soit réduite à un point. Dans ce dernier cas, nous caractérisons ce point M par ses coordonnées (b, c) dans le repère $(A, \mathbf{AB}, \mathbf{AC})$ (voir la figure V.2).

$$\mathbf{AM} = b\mathbf{AB} + c\mathbf{AC}, \quad (\text{V.15})$$

$$M \in \mathcal{F} \iff 0 \leq b, c \leq 1. \quad (\text{V.16})$$

À présent, l'appartenance de M à $\mathcal{L}$ s'écrit $\mathbf{SM} = l\mathbf{w}, l > 0$.

Proposition 7 (*prouvée dans l'annexe V.A du présent chapitre*). *En notant*

$$\mathbf{p} \equiv \mathbf{w} \wedge \mathbf{AC} \quad \text{et} \quad \mathbf{q} \equiv \mathbf{AS} \wedge \mathbf{AB}, \quad (\text{V.17})$$

les expressions de b, c et l sont données par

$$b = \langle \mathbf{p}, \mathbf{AS} \rangle / \langle \mathbf{p}, \mathbf{AB} \rangle, \quad (\text{V.18})$$

$$c = \langle \mathbf{q}, \mathbf{w} \rangle / \langle \mathbf{p}, \mathbf{AB} \rangle, \quad (\text{V.19})$$

$$l = \langle \mathbf{q}, \mathbf{AC} \rangle / \langle \mathbf{p}, \mathbf{AB} \rangle, \quad (\text{V.20})$$

le cas $\langle \mathbf{p}, \mathbf{AB} \rangle = 0$ correspondant à une direction $\mathcal{L}$ parallèle au plan support du triangle $\mathcal{F}$.

V.4.1.3 Coût algorithmique

Nous évaluons le coût du calcul de $\mathbf{p}_\theta(\mathbf{t})$ en considérant les coûts d'une addition et d'un test (comparaison) négligeables par rapport au coût d'une multiplication ou d'une division. Pour un polyèdre convexe, cette évaluation nécessite le calcul de q produits vectoriels, $2q$ produits scalaires et q divisions, soit $13q$ opérations flottantes («flops»). Pour un polyèdre non convexe, le coût dans le pire des cas s'établit comme suit. Dans le cas non dégénéré, il est identique à celui de q intersections rayon-triangles, en négligeant le coût du tri de la liste des intersections $\{l_j, j = 1, \dots, 2k\}$ ³. Chaque intersection entre $\mathcal{L}$ et une facette triangulaire nécessite 2 produits vectoriels, 4 produits scalaires et 3 divisions, soit 27 flops. Le coût de l'évaluation de (IV.28) est finalement égal à $27q$ flops, soit environ le double du coût de la projection d'un polyèdre convexe à nombre égal de sommets.

V.4.2 Mise à jour des projections

Nous considérons maintenant la mise à jour des projections d'un polyèdre suite au déplacement d'un seul sommet (par exemple V_1). Cette opération intervient dans l'algorithme d'optimisation par blocs, mais aussi pour l'évaluation de $\nabla \mathcal{J}$, que nous traitons au § V.4.3. Pour l'algorithme par blocs, nous procédons comme dans le cas 2D. À l'itération k , nous définissons la quantité $\mathbf{p} = \mathcal{A}(\mathbf{V}^{(k-1)})$, que nous mettons à jour suite au déplacement de chaque sommet V_i . Une fois l'itération terminée, les n sommets ont été ré-estimés, et nous recalculons directement les projections de $\mathbf{V}^{(k)}$, que nous comparons avec $\mathbf{p}$. Cette procédure est donc identique à celle du tableau IV.2 (page 66), où l'effet du déplacement d'un seul sommet est résumée par le tableau V.1 et décrite ci-dessous.

Dans le cas 2D, la mise à jour s'appuie sur la décomposition d'un polygone en réunion de triangles. Cela se généralise ici par la décomposition de $\mathcal{P}$ en n_1 tétraèdres, où n_1 est le nombre de voisins de V_1 . Plus précisément, nous montrons ci-dessous que l'extraction des tétraèdres est rendue possible par la structure triangulaire des facettes.

³En effet, ce dernier est de $2k \log(2k)$ flops par un algorithme classique de type *quicksort*, où k est généralement très faible ($k < 5$).

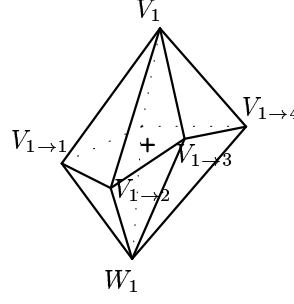


Figure V.3 – Extraction d'un mini-polyèdre $\mathcal{P}_1$ contenant V_1 et ses voisins. W_1 est un sommet fictif rajouté pour refermer $\mathcal{P}_1$.

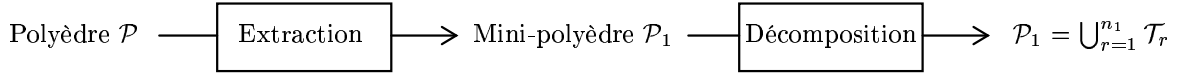


Figure V.4 – Effet du sommet V_1 sur la projection du polyèdre $\mathcal{P}$. La dépendance $\mathcal{P}(V_1)$ s'exprime uniquement à base de n_1 tétraèdres $\mathcal{T}_r$ « extraits » de $\mathcal{P}$ et ayant V_1 pour sommet.

V.4.2.1 Extraction de tétraèdres

Nous analysons l'effet du sommet V_1 sur une projection $p_{\mathbf{V}}(\mathbf{t})$ en envisageant deux transformations géométriques, dont la première est détaillée au § V.5 et illustrée par la figure V.3.

1. L'extraction d'un mini-polyèdre $\mathcal{P}_1 \subset \mathcal{P}$, dont les sommets sont V_1 , ses voisins $V_{1 \rightarrow 1}, \dots, V_{1 \rightarrow n_1}$, et enfin un sommet fictif W_1 qui sert à la fermeture de $\mathcal{P}_1$ ⁴.
2. La décomposition de $\mathcal{P}_1$ en une partition de tétraèdres $\mathcal{T}_r$ (cette étape généralise la décomposition en triangles du cas 2D). Comme $(V_1, V_{1 \rightarrow r}, V_{1 \rightarrow r+1})$, $r \in \{1, \dots, n_1\}$ sont les facettes de $\mathcal{P}$ adjacentes à V_1 , nous définissons les n_1 tétraèdres $\mathcal{T}_r$ par les sommets $(V_1, V_{1 \rightarrow r}, V_{1 \rightarrow r+1}, W_1)$.

Ces deux transformations, schématisées par la figure V.4, se matérialisent géométriquement par les relations

$$\mathcal{P} = \mathcal{P}_1^\perp + \varepsilon_1 \mathcal{P}_1 \quad \text{et} \quad \mathcal{P}_1 = \bigcup_{1 \leq r \leq n_1} \mathcal{T}_r, \quad (\text{V.21})$$

où $\mathcal{P}_1^\perp$ est un polyèdre qui contient W_1 et tous les sommets de $\mathcal{P}$ sauf V_1 , et $\varepsilon_1 = \pm 1$ renseigne le caractère extérieur ou intérieur de V_1 dans $\mathcal{P}$ (cette dernière notion est définie formellement au § V.5). Comme les tétraèdres $\mathcal{T}_r$ sont disjoints deux à deux, nous déduisons de (V.21) :

$$p_{\mathbf{V}}(\mathbf{t}) = p_{\mathcal{P}_1^\perp}(\mathbf{t}) + \varepsilon_1 \sum_{r=1}^{n_1} p_{\mathcal{T}_r}(\mathbf{t}), \quad (\text{V.22})$$

où le premier terme est indépendant du sommet V_1 . Par conséquent, la mise à jour de $p_{\mathbf{V}}(\mathbf{t})$ après déplacement de V_1 s'effectue simplement à base de projections des « anciens » et « nouveaux » tétraèdres $\mathcal{T}_r$. Ces projections sont analytiques en les sommets de $\mathcal{T}_r$, nous renvoyons le lecteur à l'annexe D pour de plus amples précisions sur leur évaluation.

⁴ W_1 est défini dans le même ordre d'idées que la direction fictive $\boldsymbol{\tau}_1$ qui intervient dans la pseudo tétraédration du cône $\Gamma_1(\mathcal{P})$ pour la définition de l'angle solide $A_1(\mathbf{V})$ (§ V.2.3).

Évaluation de $\varepsilon_i^{(k-1)}$, $\varepsilon_i^{(k)}$, caractère extérieur de $V_i^{(k-1)}$ et $V_i^{(k)}$ dans $\mathcal{P}$, selon la définition 3.
 Pour $r = 1, \dots, n_i$,
 Former les tétraèdres $\mathcal{T}_r^{(k-1)}$ et $\mathcal{T}_r^{(k)}$ à partir de $V_i^{(k-1)}$ et $V_i^{(k)}$.
 Calculer leurs projections.
 Faire $\mathbf{p} = \mathbf{p} + \varepsilon_i^{(k)} \mathcal{A}(\mathcal{T}_r^{(k)}) - \varepsilon_i^{(k-1)} \mathcal{A}(\mathcal{T}_r^{(k-1)})$.
 Fin Pour.

Tableau V.1 – Procédure de mise à jour des projections $\mathbf{p}$ suite au déplacement du i -ème sommet de $V_i^{(k-1)}$ en $V_i^{(k)}$. Le test d'acceptabilité du polyèdre $\mathcal{P}^{(k)}$ est mené par l'algorithme IV.2, et repose sur la présente procédure.

Nous déduisons directement de (V.22) un test d'acceptabilité des polyèdres reconstruits pour l'algorithme de descente par blocs. Ce test est adapté de celui du tableau IV.2 pour le cas 2D. La mise à jour de $\mathbf{p}$ est détaillée par le tableau V.1.

Remarque. L'étape 2 n'est pas réellement nécessaire, nous pourrions nous contenter de projeter les mini-polyèdres $\mathcal{P}_1$ obtenus avant et après déplacement de V_1 pour mettre à jour $p_{\mathbf{V}}(\mathbf{t})$. Nous préférons effectuer cette étape à cause de la rapidité du calcul des projections des tétraèdres (opération analytique).

V.4.2.2 Coût de la mise à jour

Ce coût n'est pas chiffrable de façon exacte, car il dépend du nombre de pixels détecteurs parcourus, *i.e.*, du nombre de pixels en lesquels les projections des tétraèdres $\mathcal{T}_r$ sont non nulles. Ce nombre est inconnu, mais généralement très faible par rapport au nombre total de pixels des détecteurs. Pour un déplacement de V_1 , nous projetons deux fois n_1 tétraèdres (n_1 tétraèdres par position de V_1). En notant $\delta r_1, \dots, \delta r_m$ le nombre de pixels détecteurs en lesquels les projections des tétraèdres sont non nulles, le coût algorithmique de ces projections est de $2 \sum_{k=1}^m (2 \delta r_k + c_k)$, en vertu des calculs menés en annexe (§ D.3.4). La quantité c_k représente le coût de calcul des paramètres communs à tous les pixels détecteurs (paramètres notés α_i, β_i au § D.3.4).

La procédure de mise à jour des projections est donc simple et à coût limité, car en général, le nombre de pixel δr_k est faible par rapport au nombre total de pixels.

V.4.3 Calcul du gradient du critère

Nous déduisons directement de ce qui précède une technique d'évaluation exacte de $\nabla \mathcal{J}$. En effet, nous montrons que les dérivées partielles $\partial \mathcal{J} / \partial V_i$ dépendent directement de celles de $p_{\mathbf{V}}(\mathbf{t})$.

V.4.3.1 Principe du calcul

Clairement, l'énergie $\mathcal{J}(\mathbf{V}) = \mathcal{K}(\mathbf{V}) + \lambda \mathcal{R}(\mathbf{V})$ fait intervenir tous les sommets de la même façon, il suffit donc d'évaluer sa dérivée partielle par rapport à V_1 (lorsqu'elle existe) pour pouvoir en déduire son gradient. Le calcul de $\partial \mathcal{R}(\mathbf{V}) / \partial V_1$ est relativement simple car $\mathcal{R}(\mathbf{V})$ s'exprime de façon analytique. Il est traité dans les annexes D et E, et nous ne considérons ici que l'évaluation de $\partial \mathcal{K}(\mathbf{V}) / \partial V_1$. Nous rappelons que l'énergie d'attache aux données est définie par :

$$\mathcal{K}(\mathbf{V}) = \sum_{k=1}^m \mathcal{K}_k(\mathbf{V}) = \sum_{k=1}^m \sum_{i,j=1}^{r_x, r_y} [\mathcal{A}_{ij}^k(\mathbf{V}) - d_{ij}^k]^2, \quad (\text{V.23})$$

où $\mathbf{d}_k = \{d_{ij}^k\}$ est la k -ème image donnée, et $\{\mathcal{A}_{ij}^k(\mathbf{V})\}$ la k -ème projection du polyèdre. En supposant que chaque projection $\mathcal{A}_{ij}^k(\mathbf{V})$ admette une dérivée partielle par rapport à V_1 , l'expression de la dérivée

partielle de $\mathcal{K}(\mathbf{V})$ est donc :

$$\partial\mathcal{K}(\mathbf{V})/\partial V_1 = 2 \sum_{k=1}^m \sum_{i,j=1}^{r_x, r_y} \partial\mathcal{A}_{ij}^k(\mathbf{V})/\partial V_1 (\mathcal{A}_{ij}^k(\mathbf{V}) - d_{ij}^k). \quad (\text{V.24})$$

Finalement, nous déduisons le résultat suivant.

Proposition 8 *De par la structure de $\mathcal{K}(\mathbf{V})$, il est suffisant de calculer la dérivée partielle d'une seule des projections par rapport à V_1 pour déduire $\partial\mathcal{K}(\mathbf{V})/\partial V_1$.*

À présent, nous nous focalisons sur une seule projection $\mathcal{A}_{ij}(\mathbf{V})$, et nous fixons la position (i, j) du pixel détecteur. Nous étudions l'existence et l'évaluation de la quantité $\partial\mathcal{A}_{ij}(\mathbf{V})/\partial V_1$. À cet effet, nous utilisons les deux transformations du § V.4.2, qui permettent de définir un mini-polyèdre $\mathcal{P}_1$ et n_1 tétraèdres $\mathcal{T}_r$ extraits de $\mathcal{P}_1$. Nous déduisons de (V.22) que :

$$\partial\mathcal{A}_{ij}(\mathbf{V})/\partial V_1 = \varepsilon_1 \partial\mathcal{A}_{ij}(\mathcal{P}_1)/\partial V_1 = \varepsilon_1 \sum_r \partial\mathcal{A}_{ij}(\mathcal{T}_r)/\partial V_1. \quad (\text{V.25})$$

Nous obtenons donc une expression analytique de $\partial\mathcal{A}_{ij}(\mathbf{V})/\partial V_1$ en fonction des sommets $V_1, \{V_{1 \rightarrow r}\}, W_1$. La dérivée partielle de $\mathcal{K}$ par rapport à V_1 s'exprime finalement par :

$$\partial\mathcal{K}(\mathbf{V})/\partial V_1 = 2\varepsilon_1 \sum_{k=1}^m \sum_{r=1}^{n_1} \sum_{i,j=1}^{r_x, r_y} \partial\mathcal{A}_{ij}^k(\mathcal{T}_r)/\partial V_1 (\mathcal{A}_{ij}^k(\mathbf{V}) - d_{ij}^k). \quad (\text{V.26})$$

Nous renvoyons enfin le lecteur au § D.3 de l'annexe D pour l'étude de la projection d'un tétraèdre et de sa différenciation.

V.4.3.2 Conclusions

Tout d'abord, précisons que le calcul ci-dessus est exact bien que non analytique. En effet, l'expression (V.26) fait intervenir les projections de $\mathcal{P}$, et dépend du sommet fictif W_1 . En revanche, si nous supposons la projection de $\mathcal{P}$ calculée, il s'agit simplement de calculer les dérivées partielles des projections de tétraèdres par rapport à leurs sommets, qui s'expriment de façon simple.

Dans l'optique de la minimisation de $\mathcal{J}$, nous exprimons en annexe une condition nécessaire et suffisante sur $\mathbf{V}$ pour l'existence de $\nabla\mathcal{J}$ (voir la proposition 10 page 159). En particulier, nous montrons que $\nabla\mathcal{J}(\mathbf{V})$ est défini si toutes les projections des arêtes du polyèdre ainsi que des arêtes fictives dues à l'introduction des sommets W_i ne contiennent pas le centre de l'un des pixels détecteurs (condition suffisante). Il existe donc une infinité de configurations en lesquelles $\mathcal{J}$ n'est pas différentiable. Néanmoins, l'ensemble de ces configurations est de mesure nulle, et pour l'une d'elles, il existe des variables par rapport auxquelles $\mathcal{J}$ possède une dérivée partielle. En d'autres termes, si l'une des projections du i -ème sommet de $\mathbf{V}$ coïncide avec le centre d'un pixel détecteur, $\mathcal{J}$ n'est pas dérivable par rapport à la V_j pour $j = i$ ou j voisin de i , mais l'est en général par rapport aux autres sommets.

Le coût de calcul de $\nabla\mathcal{K}$ (la partie complexe de $\nabla\mathcal{J}$) est difficile à chiffrer précisément car chacune des dérivées partielles $\partial\mathcal{K}/\partial V_i$ ne dépend que des pixels détecteurs dont le centre se trouve dans l'enveloppe convexe des projetés de V_i et de ses voisins. En notant $\delta r_k(i)$ ce nombre de pixels pour la k -ème projection, le coût algorithmique de $\partial\mathcal{K}/\partial V_i$ est majoré par $n_i \sum_{k=1}^m (4\delta r_k(i) + c_k(i))$, où $c_k(i)$ représente le nombre d'opérations élémentaires faites une fois pour toute (avant le parcours des pixels détecteurs) pour la projection d'un tétraèdre. Cette majoration provient de la formulation (V.26) et des conclusions tirées au § D.3.4. En pratique, le coût de calcul de $\nabla\mathcal{K}$ (la somme des quantités exprimées ci-dessus) est du même ordre de grandeur, voire plus faible, que celui de $\mathcal{K}$.

V.5 Extraction de mini-polyèdres

Cette partie fait référence à la remise à jour des projections du polyèdre suite au déplacement de l'un de ses sommets, par exemple V_1 , et au calcul de la dérivée partielle $\partial\mathcal{J}(\mathbf{V})/\partial V_1$. Nous avons vu aux § V.4.2 et V.4.3 que ces opérations nécessitent l'extraction et la projection d'un mini-polyèdre $\mathcal{P}_1$ à facettes triangulaires. Nous examinons ici les façons de définir $\mathcal{P}_1$, qui doit contenir V_1 , ses n_1 voisins, et les facettes de $\mathcal{P}$ qui lui sont adjacentes. Pour la clarté de l'exposé, nous renumérotions les voisins de V_1 par V_i , $i = 2, \dots, n_1 + 1$ et les facettes adjacentes à V_1 par $\mathcal{F}_1, \dots, \mathcal{F}_{n_1}$, par ordre d'adjacence, c'est-à-dire que :

$$\forall i \in \{1, \dots, n_1\}, \mathcal{F}_i = (V_1, V_{i+1}, V_{2+i[n_1]}) \text{ est une facette de } \mathcal{P}.$$

Une fois toutes ces facettes latérales définies, le problème est de « refermer » $\mathcal{P}_1$, c'est-à-dire de transformer la « base » $(V_2, \dots, V_{n_1+1})$ en un ensemble de facettes (voir la figure V.6). Nous verrons que cette procédure dépend du positionnement de V_1 et de la « régularité » de la base. Après avoir défini les notions de sommet « intérieur » et « extérieur » à $\mathcal{P}$ (§ V.5.1), nous caractérisons au § V.5.2 la dépendance de $\mathcal{P}$ par rapport à $\mathcal{P}_1$ de façon analogue au § IV.4.3 dans le cas 2D. Nous montrons en outre au § V.5.3 que la fermeture de $\mathcal{P}_1$ nécessite en général le rajout d'un point fictif W_1 , dont le choix est critique au sens où $\mathcal{P}_1$ doit être acceptable (sans auto-intersection).

V.5.1 Positionnement des sommets

Nous caractérisons le sommet V_1 comme « extérieur » ou « intérieur » à $\mathcal{P}$. À cet effet, nous rappelons que chaque facette $\mathcal{F}_i = (V_{i1}, V_{i2}, V_{i3})$ de $\mathcal{P}$ est définie de telle sorte que la normale

$$\mathbf{f}_i = \frac{\mathbf{V}_{i1}\mathbf{V}_{i2} \wedge \mathbf{V}_{i1}\mathbf{V}_{i3}}{\|\mathbf{V}_{i1}\mathbf{V}_{i2}\| \|\mathbf{V}_{i1}\mathbf{V}_{i3}\|} \quad (\text{V.27})$$

soit orientée vers l'extérieur⁵. Avec un tel choix, les facettes adjacentes à V_1 induisent un polygone $\mathcal{Z}_1 = \{V_i, i = 2, \dots, n_1 + 1\}$ de $\mathbb{R}^3$ orienté « vers l'extérieur », selon l'illustration de la figure V.5. C'est-à-dire que :

- Si le polygone $\mathcal{Z}_1$ est plan et convexe, $\mathbf{V}_{i-1}\mathbf{V}_i \wedge \mathbf{V}_i\mathbf{V}_{i+1}$ est orienté vers l'extérieur.
- Si le polygone $\mathcal{Z}_1$ est plan et non convexe, mais étoilé par rapport à un point G_1 , alors pour tout i , $\mathbf{G}_1\mathbf{V}_i \wedge \mathbf{G}_1\mathbf{V}_{i+1}$ est orienté vers l'extérieur.

Compte tenu de ces remarques, nous définissons de façon plus formelle le positionnement de V_1 , en utilisant le barycentre V_1^* de ses n_1 voisins.

Définition 3 *Le sommet V_1 est dit extérieur (resp. intérieur) à $\mathcal{P}$ si et seulement si $\langle \mathbf{V}_1\mathbf{V}_1^*, \mathbf{n}_1 \rangle \leq 0$ (resp. ≥ 0), où $\mathbf{n}_1$ est un vecteur « normal »⁶ à la base, défini par :*

$$\mathbf{n}_1 = \sum_{i=1}^{n_1} \mathbf{h}_i, \quad \text{où } \mathbf{h}_i = \mathbf{V}_1^*\mathbf{V}_{i+1} \wedge \mathbf{V}_1^*\mathbf{V}_{2+i[n_1]}. \quad (\text{V.28})$$

Cette définition est très proche de celle qu'utilisent Bulpitt et Efford (1996) pour la segmentation d'images 3D par maillages triangulaires.

⁵Le polyèdre $\mathcal{P}$ ainsi défini, cette propriété d'orientation de la normale reste valable quels que soient les déplacements de ses sommets (voir la figure V.5 pour l'orientation de $\mathbf{f}_1$ dans le cas d'une facette intérieure ou extérieure).

⁶En fait, la base n'est en général pas plane, mais le terme « normal » correspond bien à la notion physique de $\mathbf{n}_1$.

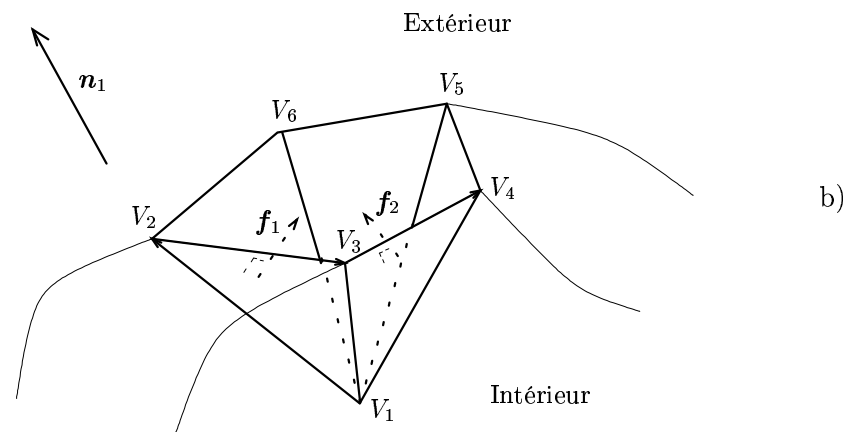
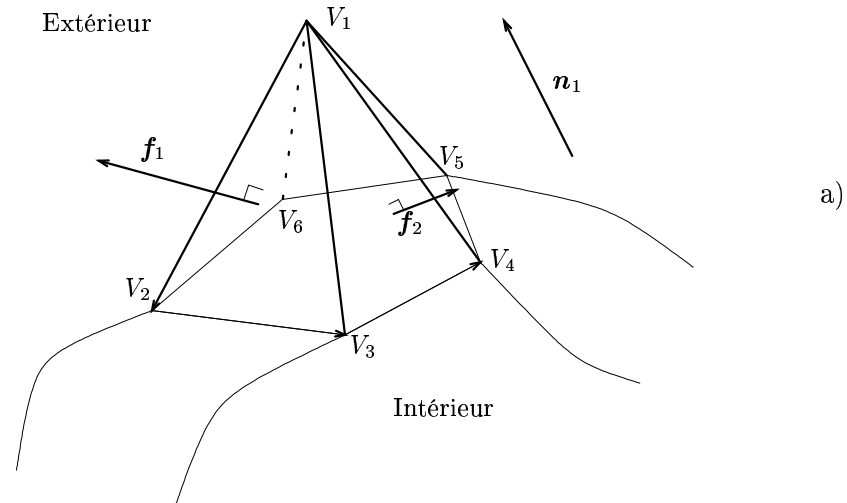


Figure V.5 – Positionnement du sommet V_1 . a) Sommet «extérieur», b) Sommet «intérieur». Dans les deux cas, les normales $\mathbf{f}_i$ aux facettes de $\mathcal{P}$ sont toujours orientées vers l'extérieur.

V.5.2 Décomposition d'un polyèdre

Nous avons vu au § IV.4.3 qu'un polygone $\mathbf{v}$ de $\mathbb{R}^2$ se décompose sous la forme $\varepsilon_1 \mathbf{t}_1 + \cup_{i \in I} \mathbf{v}_i - \cup_{j \in J} \mathbf{v}_j$, où $\mathbf{t}_1$ représente le triangle (v_n, v_1, v_2) , les polygones $\{\mathbf{v}_i\}$ et $\{\mathbf{v}_j\}$ sont formés des sommets de $\mathbf{v}$ autres que v_1 ⁷, et ε_1 vaut 1 si v_1 est extérieur à $\mathbf{v}$, -1 sinon. Ce raisonnement est toujours valable dans le cas 3D, et mène à une décomposition analogue :

$$\mathcal{P} = \varepsilon_1 \mathcal{P}_1 + \cup_{i \in I} \mathcal{P}_i - \cup_{j \in J} \mathcal{P}_j, \quad (\text{V.29})$$

où les polyèdres $\{\mathcal{P}_i\}$, $\{\mathcal{P}_j\}$ sont formés à partir des sommets de $\mathcal{P}$ autres que V_1 , et le caractère extérieur ou intérieur de V_1 n'intervient qu'au niveau du facteur $\varepsilon_1 = \pm 1$.

La grande différence avec le cas 2D réside dans la fermeture de $\mathcal{P}_1$. En 2D, elle est immédiate car le polygone extrait de $\mathbf{v}$ est un triangle. En 3D, $\mathcal{P}_1$ s'appuie sur un polygone $\mathcal{Z}_1$ volumique et en général non plan. Il existe des configurations simples où $\mathcal{Z}_1$ possède une structure convexe, *i.e.*, sa projection orthogonale suivant $\mathbf{n}_1$ est un polygone convexe de $\mathbb{R}^2$. Dans ce cas, illustré en figure V.6, la triangulation de $\mathcal{Z}_1$ en $\{\mathbf{t}_i, i = 1, \dots, n_1 - 2\}$ s'obtient de façon immédiate sans rajout de sommet fictif. De plus, la décomposition de $\mathcal{P}_1$ en la réunion de tétraèdres s'obtient facilement : $\mathcal{P}_1 = \cup \{(V_1, \mathbf{t}_i), 1 \leq i \leq n_1 - 2\}$. La plupart des mini-polyèdres ne peuvent pas être refermés comme indiqué ci-dessus, car leur base $\mathcal{Z}_1$ ne possède pas cette structure convexe. Nous sommes donc amenés à introduire un sommet fictif W_1 .

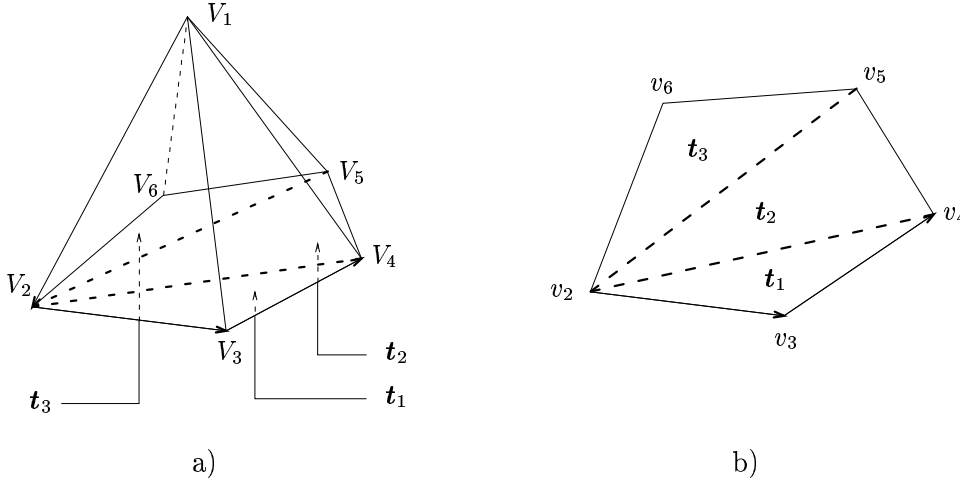


Figure V.6 – Fermeture de $\mathcal{P}_1$ dans le cas où $\mathcal{Z}_1$ possède une structure convexe. a) Mini-polyèdre extrait. b) Projection orthogonale de $\mathcal{Z}_1$ suivant $\mathbf{n}_1$, et triangulation de ce polygone convexe.

V.5.3 Fermeture d'un mini-polyèdre

L'idée naturelle, qui fonctionne dans les cas où la base a une structure quasi-planaire (plus précisément dans tous les cas où la projection orthogonale de $\mathcal{Z}_1$ suivant $\mathbf{n}_1$ est un polygone étoilé par rapport au projeté v_1^* de V_1^*) est d'utiliser le barycentre V_1^* comme sommet fictif. Nous pouvons alors trianguler $\mathcal{Z}_1$ par n_1 triangles de sommet V_1^* , comme illustré en figure V.7 (a). Si la projection de $\mathcal{Z}_1$ suivant $\mathbf{n}_1$ n'est pas étoilée par rapport à v_1^* , comme dans l'exemple de la figure V.7 (b), nous proposons de refermer $\mathcal{P}_1$ par un sommet fictif W_1 de la droite $(V_1 V_1^*)$, situé « au delà » de tous les sommets $V_i, i = 2, \dots, n_1 + 1$. En d'autres termes, W_1 est défini par $W_1 = V_1 + t_1 \mathbf{V}_1 \mathbf{V}_1^*$, où le scalaire $t_1 > 0$ vérifie :

⁷Plus précisément, ces polygones sont « distincts » deux à deux, au sens où $\forall k, l \in I \cup J, k \neq l, \text{Int}(\mathbf{v}_k \cap \mathbf{v}_l) = \emptyset$.

$$t_1 \| \mathbf{V}_1 \mathbf{V}_1^* \|^2 > \max \{ \langle \mathbf{V}_1 \mathbf{V}_1^*, \mathbf{V}_1 \mathbf{V}_i \rangle, 2 \leq i \leq n_1 + 1 \}. \quad (\text{V.30})$$

Ce procédé est illustré en figure V.8. Dans le cas (a), cas proche du cas (b) de la figure V.7, nous obtenons un mini-polyèdre sans auto-intersection⁸. L'exemple (b) montre les limites de cette stratégie lorsque V_1 est proche de V_1^* . La sélection de W_1 peut mener à un mini-polyèdre non admissible, dont l'une des arêtes (W_1, V_i) , intersecte une ou plusieurs facettes de $\mathcal{P}_1$.

Dans ce dernier cas, voire dans le cas où l'un des produits scalaires $\langle \mathbf{V}_1 \mathbf{V}_1^*, \mathbf{V}_1 \mathbf{V}_i \rangle$ est négatif, nous proposons une stratégie différente qui ne tient pas compte de la droite (V_1, V_1^*) . Nous plaçons W_1 sur la demi-droite d'extrémité V_1^* et orientée par $-\mathbf{n}_1$: $W_1 = V_1^* + t_1 \mathbf{n}_1$, où le scalaire $t_1 < 0$ vérifie :

$$t_1 \| \mathbf{n}_1 \|^2 \leq \langle \mathbf{n}_1, \mathbf{V}_1^* \mathbf{V}_i \rangle, \quad \forall i \in \{1, \dots, n_1 + 1\}. \quad (\text{V.31})$$

Nous illustrons cette stratégie en figure V.9 dans le cas de l'objet de la figure V.8 (b).

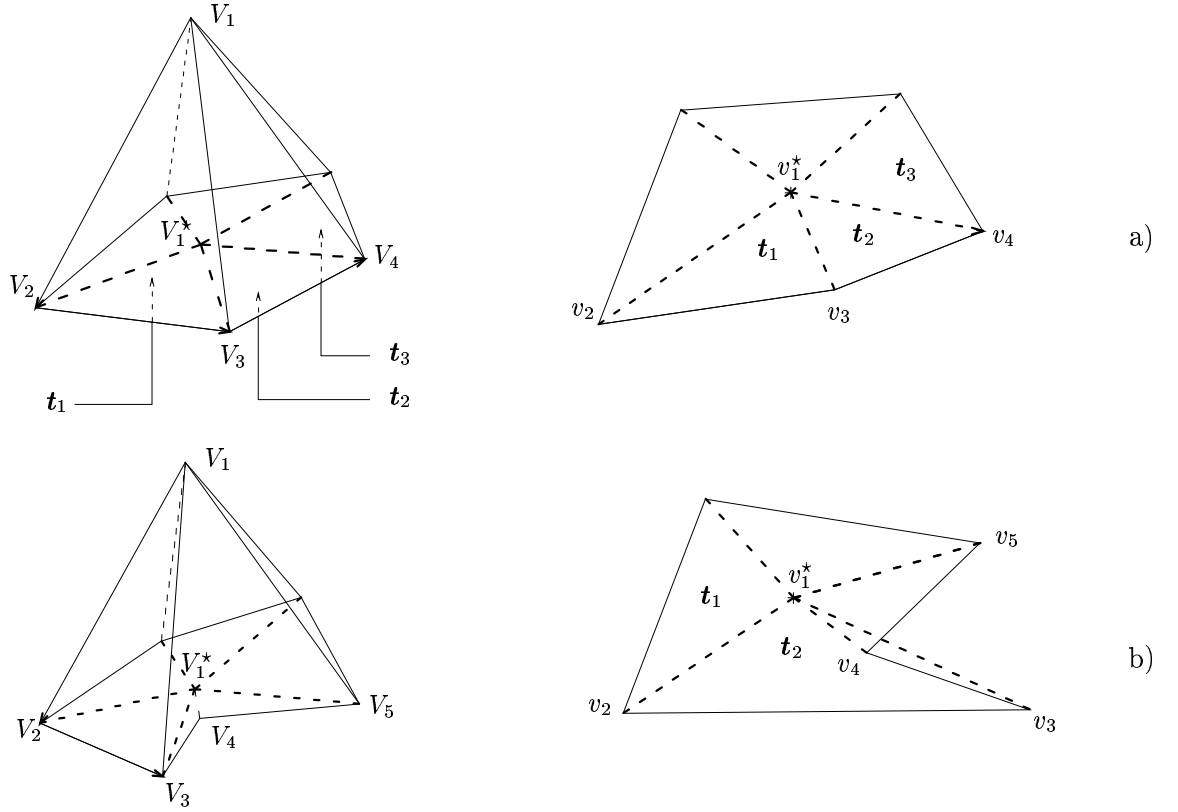


Figure V.7 – Triangulation de la base par le rajout du point fictif V_1^* et la génération de n_1 facettes triangulaires t_i . a) Cas favorable, b) Cas défavorable où la projection de la base suivant $\mathbf{n}_1$ n'est pas étoilée par rapport à v_1^* .

⁸En revanche, la décomposition de $\mathcal{P}_1$ en une partition de tétraèdres n'est pas garantie. Sur l'exemple de la figure V.8 (a), cette partition n'est pas réalisable car le tétraèdre (V_1, V_3, V_4, W_1) n'est pas inclus dans $\mathcal{P}_1$. Il faut alors projeter directement $\mathcal{P}_1$ pour mettre à jour les projections de $\mathcal{P}$.

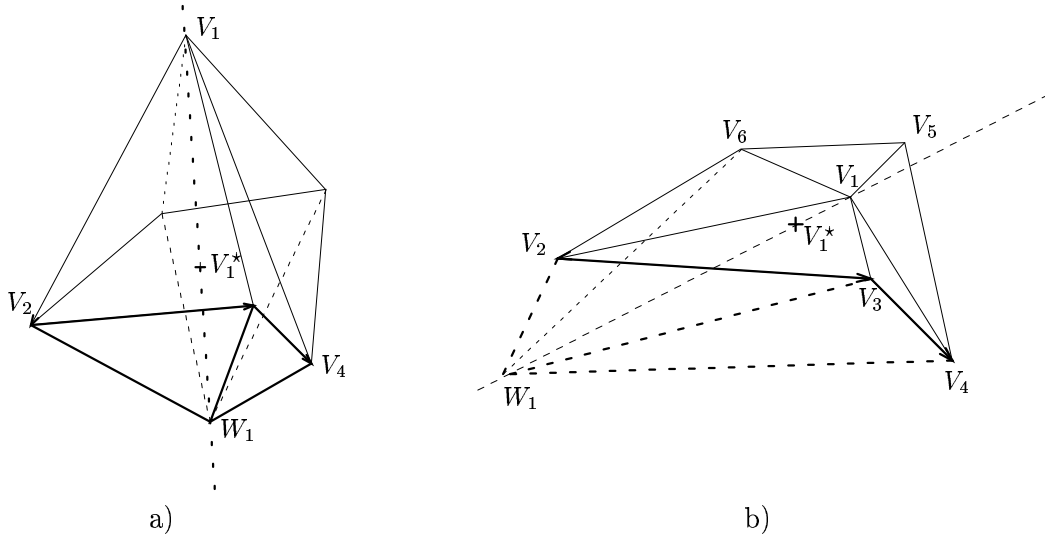


Figure V.8 – Fermeture de $\mathcal{Z}_1$ par le rajout d'un point fictif W_1 de $(V_1 V_1^*)$, choisi suivant la condition (V.30). a) Cas où $\mathcal{Z}_1$ est quasi-planaire, mais non étoilée par rapport à V_1^* , b) Cas où V_1 et V_1^* sont très proches. La droite (V_1, V_1^*) est mal définie, et la sélection de W_1 sur cette droite peut mener à une auto-intersection dans $\mathcal{P}_1$. Sur le graphe (b), le segment (W_1, V_5) intersecte une facette de $\mathcal{P}_1$: le triangle (V_1, V_6, V_2) .

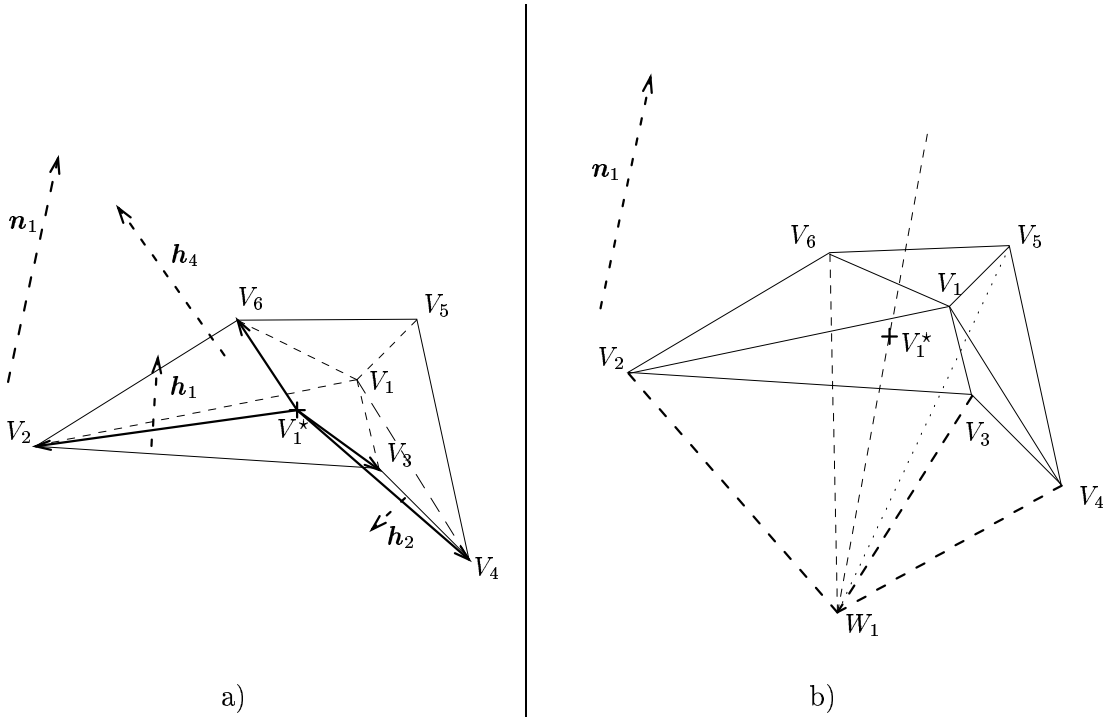


Figure V.9 – Cas où V_1 et V_1^* sont proches. Le point fictif W_1 est situé sur la demi-droite passant par V_1^* et dirigée par $-\mathbf{n}_1$. a) Visualisation des normales $\mathbf{h}_i$ à chaque triangle (V_1, V_i, V_{i+1}) et de la normale $\mathbf{n}_1$ en V_1 , b) Définition de W_1 , fermeture du mini-polyèdre $\mathcal{P}_1$.

V.5.4 Algorithme d'extraction d'un mini-polyèdre

La stratégie finale que nous appliquons pour générer le mini-polyèdre $\mathcal{P}_1$ est décrite dans le tableau V.2. Dans tous les cas, $\mathcal{P}_1$ est formé de $n_1 + 2$ sommets et $2n_1$ facettes. Ses sommets sont $V_1, V_2, \dots, V_{n_1+1}$ et W_1 , et ses facettes sont formées par les triangles $(V_1, V_{i+1}, V_{2+i[n_1]})$ et $(W_1, V_{i+1}, V_{2+i[n_1]})$, $i = 1, \dots, n_1$.

```

Calculer :
— les vecteurs  $\mathbf{h}_i$  selon (V.28) pour  $i = 1, \dots, n_1$ , en déduire  $\mathbf{n}_1 = \sum_i \mathbf{h}_i$ ;
— les produits scalaires  $p_i = \langle \mathbf{V}_1 \mathbf{V}_1^*, \mathbf{V}_1 \mathbf{V}_i \rangle$ ,  $i = 2, \dots, n_1 + 1$ .

En déduire  $\varepsilon_1 = -\text{sgn}(\langle \mathbf{V}_1 \mathbf{V}_1^*, \mathbf{n}_1 \rangle)$ .    /* Nature de  $V_1$  */

Si ( $\forall i, p_i > 0$ ),
    /* Cas régulier, rajout du sommet fictif sur  $(V_1 \mathbf{V}_1^*)$  */
    Faire  $W_1 = V_1 + \max(\{1\} \cup \{p_i / \|\mathbf{V}_1 \mathbf{V}_1^*\|^2\}) \mathbf{V}_1 \mathbf{V}_1^*$ .
Sinon,
    /* Cas singulier,  $W_1$  est généré sur  $[V_1^*, -\mathbf{n}_1)$  */
    Faire  $t_1 = 2 \min\{\langle \mathbf{n}_1, \mathbf{V}_1^* \mathbf{V}_i \rangle, 1, \dots, n_1 + 1\} / \|\mathbf{n}_1\|^2$ .    /*  $t_1 < 0$  */
    Faire  $W_1 = \mathbf{V}_1^* + t_1 \mathbf{n}_1$ .
Fin Si.
```

Tableau V.2 – Stratégie de génération du mini-polyèdre $\mathcal{P}_1$: génération du sommet fictif W_1 .

V.6 Conclusions

Dans ce chapitre, nous avons réalisé l'extension au cas tridimensionnel du modèle polygonal. Nous avons conçu la reconstruction bayésienne d'objets polyédriques à grand nombre de sommets (et à facettes triangulaires), en estimant les positions des sommets à facettes fixées. Nous réalisons cette estimation au sens du MAP, en nous inspirant largement des développements du chapitre IV. En l'absence d'hypothèse spécifique sur la surface de l'objet, l'information *a priori* incorporée porte sur sa régularité (courbure approchée), traduite par les angles solides en ses sommets. Comme pour le cas 2D, la reconstruction mène à des problèmes à la fois théoriques (optimisation d'un critère non convexe) et de mise en œuvre (manipulation des projections du modèle). Pour le premier point, nous préconisons un algorithme déterministe d'optimisation locale (méthode de descente sur $\mathbf{V}$). Quant au second, il nécessite souvent l'extraction d'objets locaux aux sommets (mini-polyèdres), dont les projections s'expriment analytiquement.

Le modèle présenté dans ce chapitre est relativement simple au sens où il s'appuie sur un nombre fixé de sommets et de facettes. Dans la littérature de la vision de formes 3D, de nombreux travaux étudient les raffinages de maillage, et peuvent être appliqués à notre problème. En particulier, les transformations dites *eulériennes* réalisent des modifications locales qui ne changent pas la topologie du maillage. Il s'agit de rajouts de sommets, et de l'élimination de sommets et d'arêtes⁹ (Rivara, 1984; Lachaud et Montanvert, 1998). Dans le même ordre d'idées, certains chercheurs intègrent dans leur modèle des notions de singularité (zones à fortes courbures, aussi appelées *kinks*) (Schroeder et al., 1992). Par exemple, Hanson et al. (1996) définissent un modèle qui autorise ces singularités, et construisent un critère d'Akaike pour pénaliser leur nombre.

Dans (Soussen et Mohammad-Djafari, 1999a,b), nous avons envisagé un modèle à nombre variable

⁹Les rajouts de sommets sont faits dans des zones de fortes courbures, et les éliminations concernent en particulier les arêtes dont les angles diédraux (angles entre les deux facettes adjacentes à l'arête) sont importants.

de sommets, en développant une stratégie multirésolution¹⁰. Elle consiste à augmenter progressivement le nombre de sommets du polyèdre, en réalisant R reconstructions successives $\widehat{\mathcal{P}}_1, \dots, \widehat{\mathcal{P}}_R$ de plus en plus fines. Cette stratégie est du même type que celle développée au chapitre II (§ II.5) pour le modèle par voxels, nous avons pris le parti de ne pas la développer pour privilégier la lisibilité de l'exposé. Nous entendons la perfectionner dans de futurs travaux, en utilisant les techniques de remaillage.

V.A Preuve de la proposition 7

Nous déduisons de (V.15) et (V.17) les expressions des produits vectoriels de $\mathbf{AM} = \mathbf{AS} + \mathbf{SM}$ avec $\mathbf{AC}$ et $\mathbf{AB}$:

$$\mathbf{AM} \wedge \mathbf{AC} = \begin{cases} b\mathbf{AB} \wedge \mathbf{AC} \\ \mathbf{AS} \wedge \mathbf{AC} + l\mathbf{p} \end{cases} \quad \text{et} \quad \mathbf{AM} \wedge \mathbf{AB} = \begin{cases} -c\mathbf{AB} \wedge \mathbf{AC} \\ \mathbf{q} + l\mathbf{w} \wedge \mathbf{AB} \end{cases}.$$

En effectuant le produit scalaire de ces quantités avec $\mathbf{w}$, et en remarquant que

$$\begin{aligned} \langle \mathbf{AB} \wedge \mathbf{AC}, \mathbf{w} \rangle &= -\langle \mathbf{p}, \mathbf{AB} \rangle, \\ \langle \mathbf{AS} \wedge \mathbf{AC}, \mathbf{w} \rangle &= -\langle \mathbf{p}, \mathbf{AS} \rangle, \\ \langle \mathbf{AS} \wedge \mathbf{AC}, \mathbf{AB} \rangle &= -\langle \mathbf{q}, \mathbf{AC} \rangle, \end{aligned}$$

nous obtenons le résultat. □

¹⁰Ce modèle permet en outre une réduction sensible de la charge calculatoire.

CHAPITRE VI

SIMULATIONS

- VI.1 Introduction
- VI.2 Approche par voxels
- VI.3 Approche par contours paramétriques globaux
- VI.4 Approche par splines d'ordre un
- VI.5 Conclusions

VI.1 Introduction

Le but de ce chapitre est d'illustrer et de comparer les performances des méthodes présentées dans ce manuscrit. À cet effet, nous traitons plusieurs jeux de données synthétiques par les différentes classes de méthodes. Dans plusieurs sections de ce chapitre, nous commencerons par traiter le cas de données 2D lorsqu'il permet une meilleure illustration des méthodes. Avant de procéder à ces simulations, nous définissons les jeux de données sur lesquels nous évaluons l'ensemble des méthodes (§ VI.1.1). Puis, nous formalisons les critères de qualité des reconstructions (§ VI.1.2).

VI.1.1 Nature des données

VI.1.1.1 Données 2D

Le jeu de données 2D que nous traitons de façon poussée est noté (I), il est représenté sur la figure VI.1 (I). Il s'agit de reconstruire une forme $\mathcal{C}^*$ (forme synthétique définie par un polygone à 40 sommets) fortement non convexe à partir de quatre projections parallèles à angles limités. Les angles θ_k de ces projections sont équirépartis dans l'intervalle $(0, \pi/2)$, et chaque projection contient $r = 64$ données. Enfin, le rapport signal à bruit, défini par

$$\text{RSB} = 10 \log(v_p/v_b), \quad (\text{VI.1})$$

où v_p et v_b sont respectivement les variances empiriques des projections non bruitées de $\mathcal{C}^*$ et du bruit, est fixé à 10 dB. Sur cet exemple simple, nous illustrons les performances des méthodes par contours paramétriques, et notamment de la méthode polygonale (§ VI.3.2 et VI.4.1). Après avoir vérifié le caractère mal posé de la reconstruction de polygones fermés, nous évaluons l'approche régularisée présentée au chapitre IV. Nous étudions les influences du nombre de sommets, du paramètre λ de régularisation, et nous montrons la dépendance du résultat à l'initialisation.

Le jeu de données de la figure VI.1 (II) est utilisé ponctuellement pour illustrer nos choix concernant la méthode par contours harmoniques (cf. § VI.3.2). Il est clair que c'est un objet complexe qui ne possède pas des propriétés de douceur locale, le but n'est donc pas de traiter ce cas par l'approche polygonale. Les angles de projections sont toujours équirépartis sur $(0, \pi/2)$, et le rapport signal à bruit est de 20 dB.

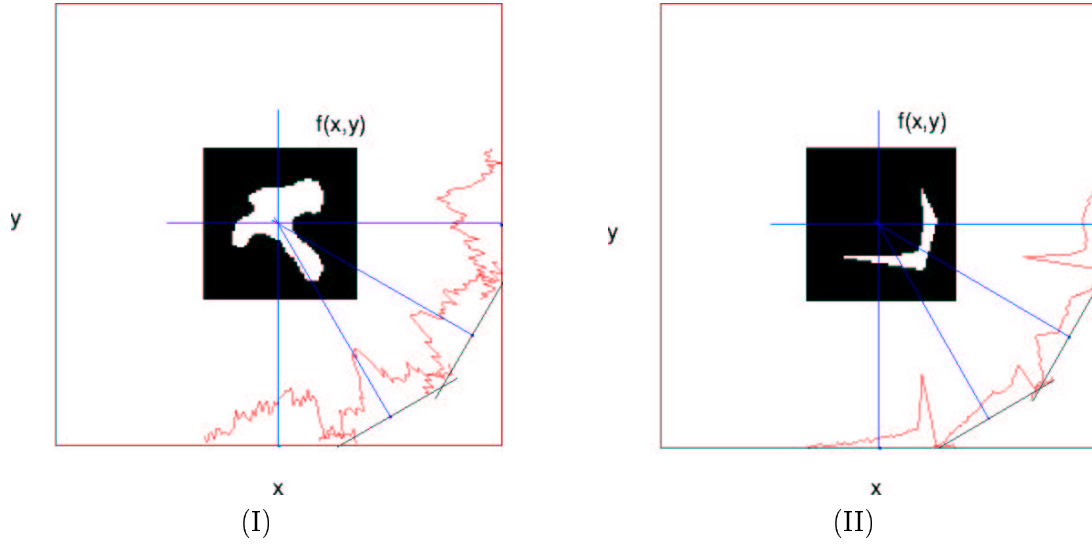


Figure VI.1 – Données synthétiques 2D. (I) Représentation de l'objet $\mathcal{C}^*$ et de ses quatre projections (RSB = 10 dB). Ce sont les données traitées dans la suite de ce chapitre. (II) Représentation du second objet et de ses projections (RSB = 20 dB). Nous ne traitons ces données que pour illustrer certaines propriétés du modèle harmonique (§ VI.3.2).

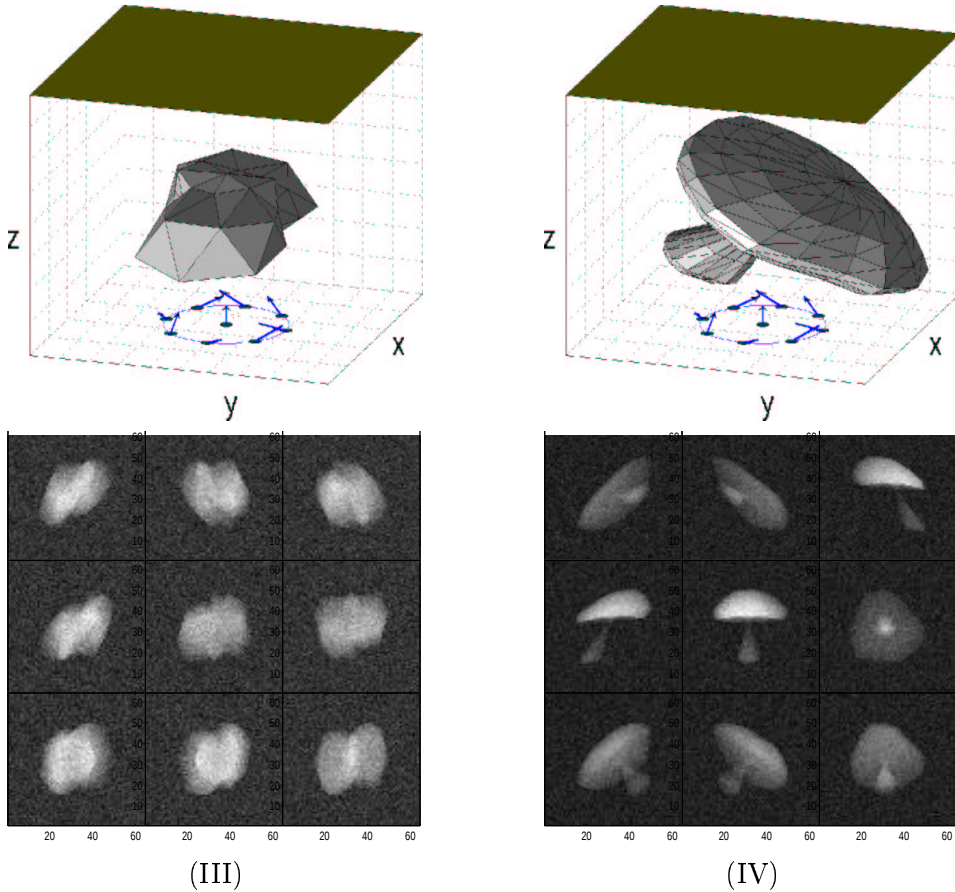


Figure VI.2 – Données synthétiques 3D. Représentation des objets $\mathcal{C}^*$ et de leurs projections (RSB = 10 dB). Les directions des projections sont également représentées. Il s'agit de 9 projections, dont les angles sont spécifiés par le tableau VI.1. Les données $\mathbf{d}_k$ sont des images de taille 64×64 disposées en lignes (les trois images de la première ligne correspondent aux trois premières projections).

i	1	2	3	4	5	6	7	8	9
θ_i	$\pi/4$	$3\pi/4$	$-\pi/4$	$5\pi/4$	$-\pi/2$	$\pi/2$	0	π	0
φ_i	φ	φ	φ	φ	$\pi/4$	$\pi/4$	$\pi/4$	$\pi/4$	0

Tableau VI.1 – Données 3D. Valeurs des coordonnées sphériques des vecteurs dirigeant les 9 projections ($\varphi = \arccos(\sqrt{3}/3)$). Le dernier est dirigé suivant z .

VI.1.1.2 Données 3D

Les données¹ 3D concernent les objets présentés en figure VI.2 (objets inclus dans le cube $(0, 1)^3$). Nous avons simulé neuf projections parallèles bruitées sur le même plan détecteur, d'équation $z = 1.5$, suivant les angles décrits en coordonnées sphériques sur le tableau VI.1. Chaque projection est une image bidimensionnelle de taille 64×64 . Tous les angles φ_j sont en valeur absolue inférieurs à $\pi/4$, ce qui limite la quantité d'information apportée par les projections (nous manquons d'information sur l'objet suivant les directions orthogonales à l'axe z). De plus, le rapport RSB est de 10 dB. Les objets (III) et (IV) ont des topologies assez différentes. Ils sont tous deux non convexes, mais le second présente davantage d'irrégularités spatiales. Plus précisément, la modélisation polyédrique de (IV) nécessite un grand nombre de sommets (cet objet est défini par un polyèdre à 226 sommets et 448 triangles). D'autre part, ses axes principaux d'inertie ne rendent pas compte de la pseudo-symétrie de l'objet (nous reviendrons sur ce point au § VI.3.2). Ces éléments rendent complexe la reconstruction de $\mathcal{C}^*$, et permettent d'explorer les limites des méthodes de reconstruction.

VI.1.2 Qualité des reconstructions

Il existe plusieurs mesures de qualité des reconstructions. Nous en sélectionnons deux :

1. La qualité de la carte de densité reconstruite.
2. Le coût numérique de son évaluation (en temps CPU).

Nous recherchons à minimiser conjointement ces deux quantités. En fait, nous ne tenons pas compte de la seconde dans le cas 2D, car toutes les reconstructions sont obtenues de façon rapide. Par contre, ce facteur devient significatif dans le cas 3D, car le nombre de paramètres ainsi que la taille des données induisent des coûts numériques importants.

Qualité des cartes de densités. Nous revenons à présent sur le premier facteur, en formalisant la qualité de l'image reconstruite (ou du contour reconstruit). Il s'agit de comparer $\hat{\mathbf{f}}$ (approche par voxels) ou $\hat{\mathcal{C}}$ (approche par contour) au contour réel synthétique $\mathcal{C}^*$. Pour cela, il existe plusieurs options, et notamment :

- la comparaison des projections de l'image reconstruite à celles de $\mathcal{C}^*$, par un critère du type $\|\mathcal{A}(\mathcal{C}_{\mathbf{x}}) - \mathcal{A}(\mathcal{C}^*)\|^2$, où $\mathcal{A}$ est l'opérateur de projection.
- la comparaison directe de $\hat{\mathbf{f}}$ ou $\hat{\mathcal{C}}$ à $\mathcal{C}^*$.

Nous excluons la première pour deux raisons. D'une part, nous minimisons généralement un critère régularisé, qui diffère du critère des moindres carrés (distance $\mathcal{L}_2$ entre les données et la projection $\mathcal{A}(\mathcal{C}_{\mathbf{x}})$). D'autre part, nous avons montré au chapitre I le caractère mal posé du problème de reconstruction. En particulier, dans nos conditions, nous n'avons pas l'unicité des contours qui vérifient $\mathcal{A}(\mathcal{C}_{\mathbf{x}}) = \mathcal{A}(\mathcal{C}^*)$. En d'autres termes, certains contours qui vérifient cette égalité ne sont pas satisfaisants au sens où ils ne vérifient pas les propriétés de douceur locale. Par conséquent, nous considérons la distance de l'image reconstruite $\hat{\mathbf{f}}$ ou $\hat{\mathcal{C}}$ à $\mathcal{C}^*$. Pour cela, une possibilité est d'utiliser la distance de Hausdorff (Berger, 1987) entre deux contours.

¹Le second jeu de données est également introduit en annexe B.

Définition 4 La distance de Hausdorff entre deux courbes ou surfaces $\mathcal{C}_1$ et $\mathcal{C}_2$ repose sur la notion de ε -voisinage d'un contour $\mathcal{C}$:

$$\mathcal{C}^\varepsilon = \{M, d(M, \mathcal{C}) < \varepsilon\}, \quad (\text{VI.2})$$

où $d(M, \mathcal{C})$ représente la distance euclidienne du point M au contour². Elle est définie par :

$$\mathcal{H}(\mathcal{C}_1, \mathcal{C}_2) = \inf \{\varepsilon, \mathcal{C}_1 \subset \mathcal{C}_2^\varepsilon \text{ et } \mathcal{C}_2 \subset \mathcal{C}_1^\varepsilon\}. \quad (\text{VI.3})$$

En pratique, cette distance mène au critère $\mathcal{H}(\widehat{\mathcal{C}}, \mathcal{C}^*) / \mathcal{H}(\{G\}, \mathcal{C}^*)$, où G est un point intérieur à $\mathcal{C}^*$, et le terme de normalisation $\mathcal{H}(\{G\}, \mathcal{C}^*)$ représente une mesure de la taille de $\mathcal{C}^*$.

Critère retenu. La stratégie précédente pose plusieurs problèmes. D'une part, les calculs des distances $d(M, \mathcal{C})$ s'avèrent complexes. D'autre part, les méthodes par voxels ne produisent pas un contour $\widehat{\mathcal{C}}$, mais une carte de densité $\widehat{\mathbf{f}} \in (0, 1)^n$. L'utilisation du critère (VI.3) impose alors de procéder à une segmentation de l'image reconstruite en un contour fermé. Pour toutes ces raisons, nous n'optons pas pour cette mesure. Nous lui préférons une mesure de *distance entre cartes de densités*. En d'autres termes, nous définissons à partir de $\mathcal{C}^*$ une carte de densité binaire « idéale » $\mathbf{f}^\uparrow$ d'une résolution très fine (en pratique, $\mathbf{f}^\uparrow$ est une image à $n^\uparrow = 128^3$ ou 256^3 voxels), selon le processus de discrétisation déjà décrit au chapitre IV (§ IV.4.1). Nous procédons de même avec la reconstruction $\widehat{\mathbf{f}}$ ou $\widehat{\mathcal{C}}$ en produisant l'image $\widehat{\mathbf{f}}^\uparrow$ (également à $n^\uparrow$ voxels). La distance entre $\mathbf{f}^\uparrow$ et $\widehat{\mathbf{f}}^\uparrow$ est alors

$$\mathcal{H}(\mathbf{f}^\uparrow, \widehat{\mathbf{f}}^\uparrow) = \sum_{i=1}^{n^\uparrow} |f_i^\uparrow - \widehat{f}_i^\uparrow| \quad (\text{VI.4})$$

(distance au sens de la norme $\mathcal{L}_1$). D'une façon générale, nous utiliserons les notations $\mathcal{H}(\widehat{\mathbf{f}}, \mathcal{C}^*)$ et $\mathcal{H}(\widehat{\mathcal{C}}, \mathcal{C}^*)$ pour désigner les quantités (VI.4) associées aux reconstructions par voxels ou par contours.

VI.1.3 Organisation du chapitre

L'organisation de ce chapitre suit globalement l'ordre dans lequel les méthodes ont été présentées dans le manuscrit. Nous commençons par les méthodes par voxels, en proposant des résultats mono-résolution et multirésolution (§ VI.2), avant de présenter les méthodes par contours paramétriques. Les performances de ces dernières sont illustrées aux § VI.3 et VI.4, en suivant l'ordre de complexité des modèles déjà établi au chapitre III. En d'autres termes, nous commençons par les modèles à très faible nombre de paramètres (§ VI.3.1), avant d'aborder les modèles à grands nombres de paramètres, qui nécessitent une information *a priori* structurelle moins importante sur les contours. Parmi ces derniers, nous illustrons la notion de localité en étudiant successivement les performances des modèles harmoniques (§ VI.3.2) et des modèles spline d'ordre 1 (§ VI.4).

VI.2 Approche par voxels

VI.2.1 Stratégie monorésolution

L'approche monorésolution consiste à minimiser le critère convexe

$$\mathcal{J}(\mathbf{f}) = \mathcal{K}(\mathbf{f}) + \lambda \sum_{i \sim j} \phi_T(f_i - f_j) + \mu \sum_i f_i \quad (\text{VI.5})$$

sur l'intervalle $(0, 1)^n$, où n est le nombre total de voxels de la scène (voir le § II.2.3 du chapitre II), et ϕ_T est la fonction de Huber. Sans perte de généralité, nous supposons que le nombre de voxels est

² $\mathcal{C}^\varepsilon$ est donc une bande de largeur 2ε qui entoure le contour $\mathcal{C}$.

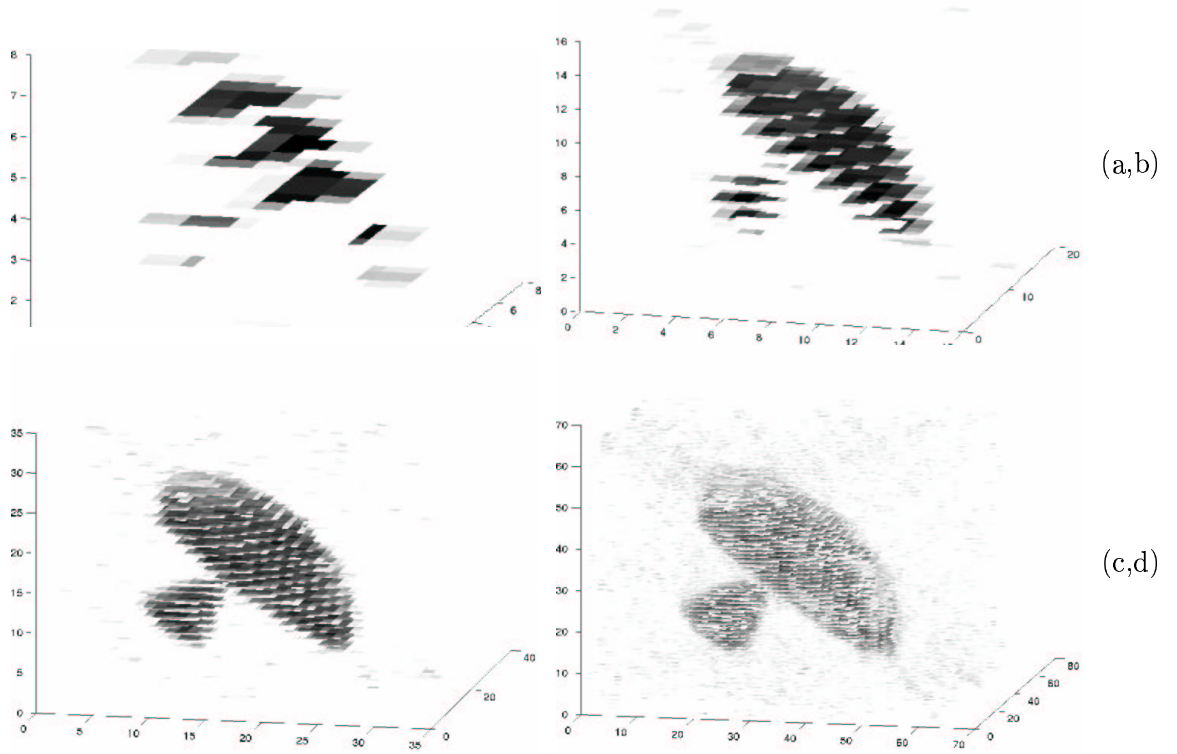


Figure VI.3 – Résultats monorésolution, pour un nombre de voxels $n = p^3$, où $p = 8, 16, 32, 64$. Les niveaux de gris de ces figures correspondent aux valeurs des densités dans l'intervalle $(0, 1)$.

cubique : $n = p^3$, où $p = 8, 16, 32, \dots$ est le nombre de voxels associés à la discrétisation de la scène suivant chacune des directions x, y , et z . Dans la suite, nous référençons p comme la *résolution* de la scène. Le critère (VI.5) étant convexe, l'initialisation $\mathbf{f}^{(0)}$ influe uniquement sur le coût numérique de l'optimisation, et non sur son résultat. Nous optons donc pour la rétroprojection des données, image de taille 16^3 voxels représentée sur la figure VI.4 (a). L'hyperparamètre μ est fixé empiriquement à $\lambda/3$, et λ est lui-même choisi de telle sorte que pour l'initialisation, $\mathcal{K}(\mathbf{f}^{(0)}) = 0.9 \mathcal{J}(\mathbf{f}^{(0)})$. Enfin, l'optimisation de $\mathcal{J}$ est menée par l'algorithme du gradient projeté décrit au chapitre II.

La figure VI.3 représente les reconstructions $\hat{\mathbf{f}} \in (0, 1)^n$ obtenues pour les résolutions $p = n^{1/3} = 8, 16, 32, 64$. Ces représentations 3D sont les empilages de p coupes de l'image « continue » $\hat{f}(x, y, z)$ suivant l'axe vertical z , où chaque coupe est une image 2D de taille $p \times p$. Dans toutes ces illustrations, nous prenons le parti de ne pas représenter les voxels de valeur nulle³. D'autre part, les images sont représentées en niveau de gris de telle sorte que les voxels les plus foncés correspondent aux valeurs proches de 1. Le grand nombre de voxels nuls des reconstructions de la figure VI.3 découle directement du domaine de minimisation. En effet, l'utilisation de l'algorithme du gradient projeté (et plus précisément l'opérateur de « projection » sur l'intervalle $(0, 1)^n$ défini au § II.3) ramène naturellement les valeurs négatives des voxels à 0. Les reconstructions obtenues sont de bonne qualité, au sens où nous retrouvons nettement la forme de l'objet synthétique $\mathcal{C}^*$. En revanche, cette approche présente plusieurs désavantages :

- son coût numérique s'avère explosif pour des résolutions spatiales très fines.
- l'encombrement mémoire qu'elle impose, par le stockage de p^3 variables scalaires (non binaires).

Nous montrons au paragraphe suivant que la technique multirésolution permet de réduire ces deux

³La notion d'empilage de coupes est plus flagrante sur la figure VI.4 (a), qui est le résultat d'une rétroprojection. En effet, aucun voxel de cette image n'est nul, et donc l'ensemble des $n = p^3$ voxels est représenté.

facteurs de façon sensible, tout en produisant des résultats d'une bonne qualité.

VI.2.2 Stratégie multirésolution

Les résultats de cette approche sont illustrés par la figure VI.4. Nous avons choisi de définir trois niveaux de résolution, en reconstruisant successivement des cartes de densité $\hat{\mathbf{f}}_1, \hat{\mathbf{f}}_2, \hat{\mathbf{f}}_3$ aux résolutions $p = 16, 32, 64$. La première reconstruction (l'estimation de $\hat{\mathbf{f}}_1$) est effectuée par la stratégie décrite au paragraphe précédent. Aux niveaux de résolution $r = 2$ et $r = 3$, nous excluons du modèle tous les voxels nuls de la scène $\hat{\mathbf{f}}_{r-1}$ pour définir la paramétrisation courante, puis nous utilisons $\hat{\mathbf{f}}_{r-1}$ pour former la solution initiale au niveau courant (voir le § II.5). Pour la simulation présentée en figure VI.4, les nombres v_r de voxels actifs (*i.e.*, le nombre de voxels dont nous estimons les valeurs, celles des autres étant fixées à 0) sont $v_2 = 5512$ et $v_3 = 32656$. Cela représente respectivement 16,8 % et 12,5 % du nombre total de voxels ($n = p^3$), au sens de la discrétisation définie au § VI.2.1.

Outre la nette diminution de l'espace mémoire nécessaire à la représentation du modèle, la stratégie multirésolution permet de ne pas visiter les zones de l'espace dans lesquelles l'approche monorésolution reconstruit des voxels à valeurs faibles mais non nulles. Les quatre reconstructions de la figure VI.4 ne diffèrent pas fondamentalement des reconstructions monorésolution. Par contre, le coût numérique de leur évaluation est nettement réduit. Cela s'explique par deux facteurs. D'une part, le nombre v_r de paramètres est nettement réduit. D'autre part, les initialisations aux résolutions $r = 2$ et $r = 3$ (ce sont les « adaptations » de $\hat{\mathbf{f}}_1$ et $\hat{\mathbf{f}}_2$) sont très bonnes, au sens où $\hat{\mathbf{f}}_r^{(0)}$ est très proche du minimiseur du critère $\mathcal{J}_r$. Nous illustrons cette remarque par la figure VI.5, qui montre la rapidité de la convergence de l'algorithme d'optimisation de $\mathcal{J}_r$: la convergence de l'algorithme du gradient aux niveaux $r = 2$ et $r = 3$ s'obtient en 3 itérations, contre de l'ordre de 20 itérations dans le cas monorésolution. Par ailleurs, nous illustrons sur la figure VI.6 la qualité des reconstructions en termes de distance à la scène réelle et de temps CPU (par itération de l'algorithme de gradient). Pour les reconstructions à 64^3 voxels, le temps CPU global des stratégies monorésolution et multirésolution (pour respectivement 15 itérations, et 10 itérations par niveau de résolution) sont de 4387 et 852 secondes. Enfin, la figure VI.6 (a) montre une meilleure qualité de reconstruction finale au sens du critère (VI.4).

Remarque. Les deux reconstructions associées à $p = 16$ sont obtenues par le même algorithme (l'algorithme monorésolution) mais pour des nombres d'itérations différents. En effet, nous avons fixé ces nombres à 15 itérations pour l'algorithme monorésolution et 10 itérations par résolution pour l'algorithme multirésolution, d'où la différence entre les deux valeurs de qualité obtenues pour $p = 16$ (figure VI.6 (a)).

VI.2.3 Conclusions

L'approche par voxels (monorésolution) permet d'obtenir des résultats de bonne qualité, mais les discrétisations fines de la scène mènent à l'estimation d'un très grand nombre de paramètres, et une mise en œuvre numériquement coûteuse. Dans ces cas, nous avons montré l'intérêt de la stratégie multirésolution, qui permet une réduction drastique de la charge calculatoire en ne considérant que certaines zones spécifiques de la scène. En particulier, des reconstructions sur trois niveaux de résolution donnent satisfaction. Toutefois, cette stratégie souffre également de l'explosion du nombre de paramètres pour des discrétisations très fines de la scène (sur 512^3 ou 1024^3 voxels).

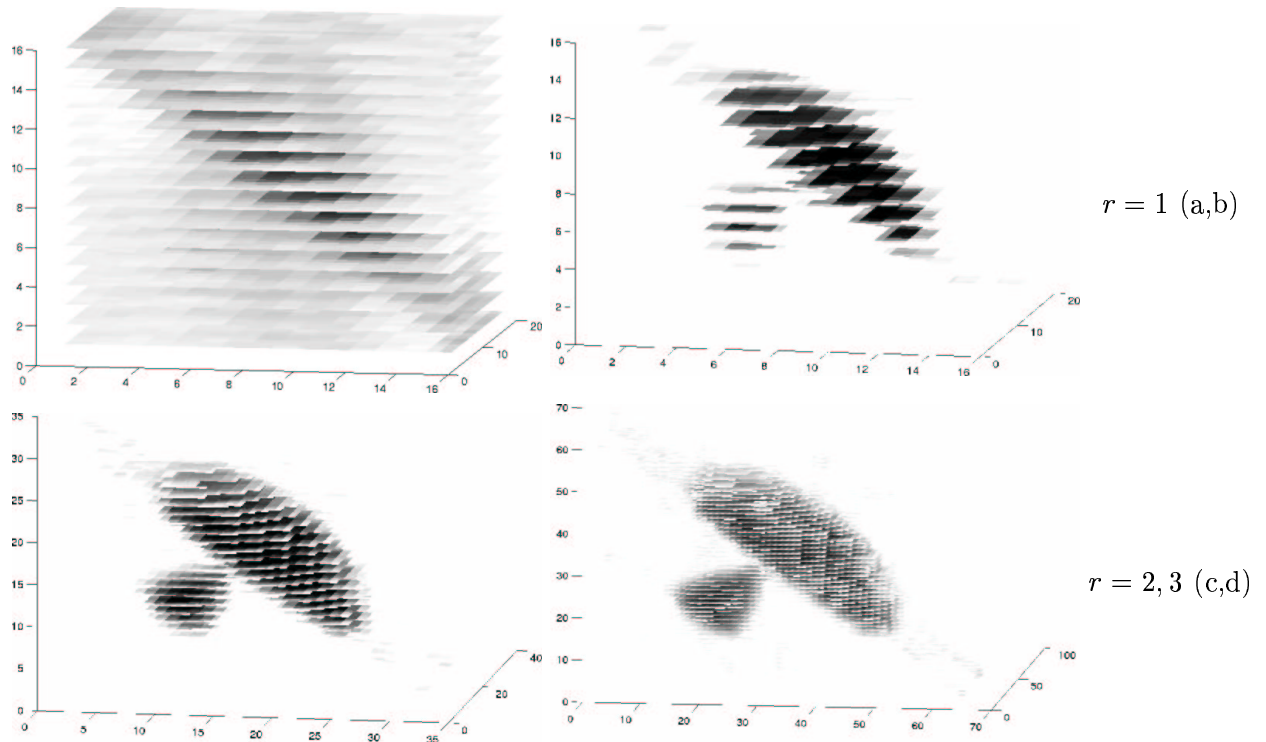


Figure VI.4 – Technique de multirésolution sur trois niveaux : reconstruction des images « creuses » $\hat{f}_1, \hat{f}_2, \hat{f}_3$, correspondant à des scènes de p^3 voxels, où $p = 16, 32, 64$ respectivement (b,c,d). L'image (a) est la rétroprojection des données. Elle comporte 16^3 voxels, et elle est utilisée pour l'initialisation au premier niveau de résolution.

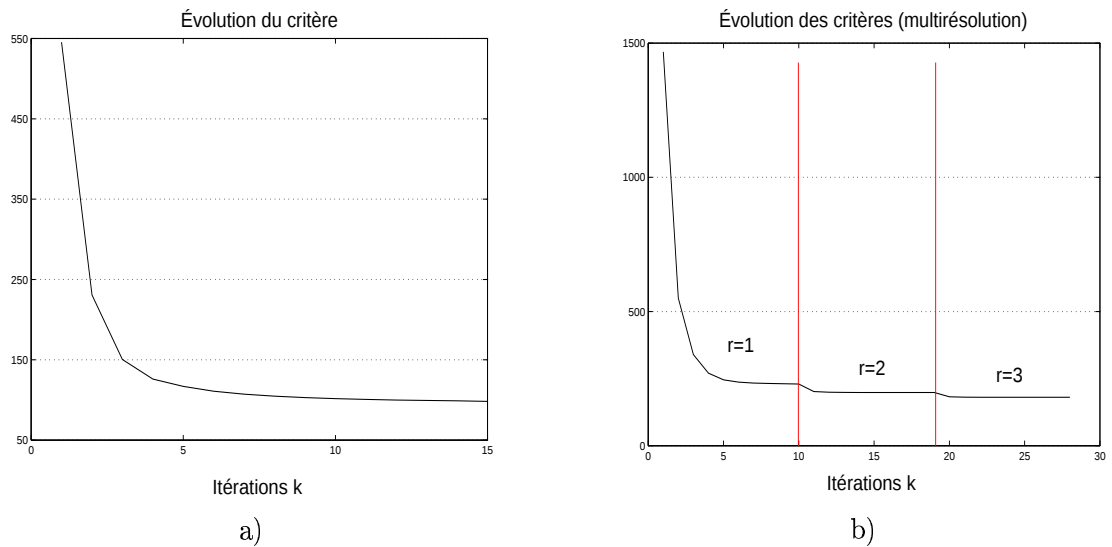


Figure VI.5 – Convergence de l'algorithme du gradient projeté pour les simulations monorésolution (a) et multirésolution (b). a) Reconstruction d'une scène de 64^3 voxels, illustrée en figure VI.3 (d). b) Reconstruction multirésolution sur trois niveaux, où le dernier niveau correspond à une scène de 64^3 voxels (figure VI.4).

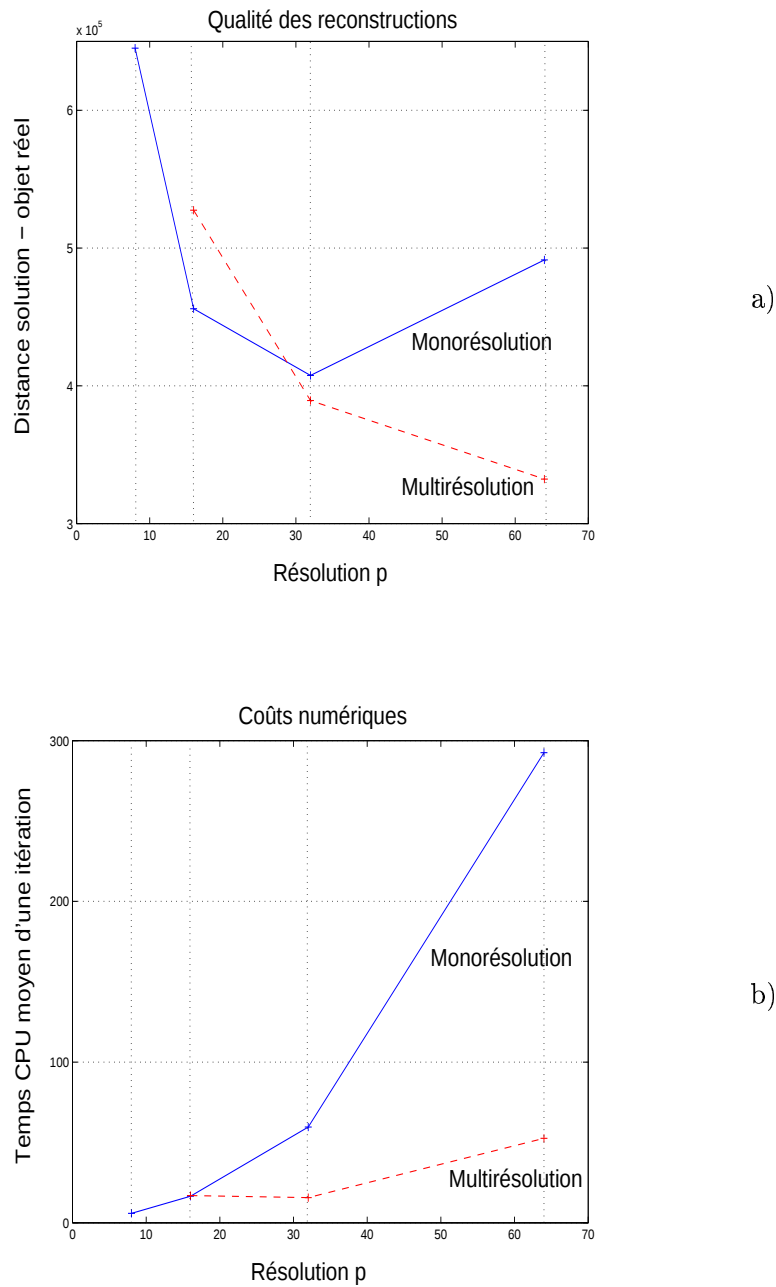


Figure VI.6 – Quantification des reconstructions en fonction de la résolution p de la scène, selon les deux critères définis au § VI.1.2. Les courbes en traits pleins et pointillés correspondent aux méthodes monorésolution et multirésolution respectivement. a) Distance de la solution $\hat{\mathbf{f}}$ (raffinée en $\hat{\mathbf{f}}^\uparrow$ sur 256^3 voxels) à la carte de densité réelle (raffinée) $\mathbf{f}^\uparrow$, b) Coût numérique en temps CPU (en secondes) *par itération*.

VI.3 Approche par contours paramétriques globaux

Nous présentons à présent les reconstructions par les modèles paramétriques globaux sélectionnés au chapitre III. Nous envisageons deux classes de modèles : les quadriques et les superquadriques (modèles à très faible nombre de paramètres), et les modèles harmoniques, à plus grand nombre de paramètres. Ils sont utilisés dans des buts différents. Les premiers permettent d'évaluer les caractéristiques principales de $\mathcal{C}^*$, alors que les seconds fournissent des informations sur ses composantes « haute fréquence ». Après avoir évoqué brièvement les premiers modèles (§ VI.3.1), nous développons les seconds, en commençant par traiter les données 2D (§ VI.3.2).

VI.3.1 Quadriques et superquadriques

Nous ne revenons pas en détail sur la méthode des moments, exposée dans l'annexe B, et préconisée pour la reconstruction d'ellipsoïdes *via* l'estimation des moments géométriques de $\mathcal{C}^*$ d'ordre inférieur ou égal à 2. Elle mène dans le cas 3D à l'ellipsoïde représenté en figure VI.7 (a) pour le jeu de données (IV). Le modèle par superquadrique ne pose pas de problème particulier, mis à part le calcul approché de ses projections. La reconstruction obtenue pour le jeu de données (IV) (figure VI.7 (b)) apporte peu d'améliorations par rapport à la reconstruction ellipsoïdale. Clairement, ces reconstructions sont des formes convexes (contrairement à l'objet réel $\mathcal{C}^*$), qui ne retiennent que la « tête » du champignon. Nous verrons au § VI.4.2 qu'elles s'avèrent parfois insuffisantes en tant qu'initialisations de la méthode polyédrique.

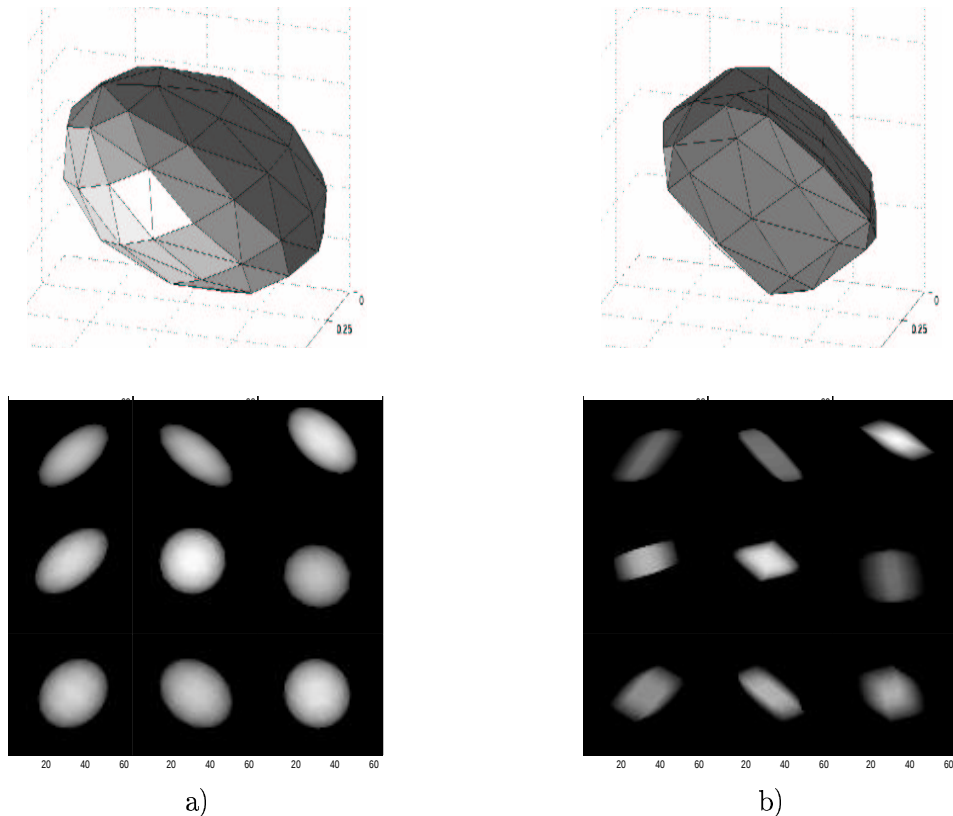


Figure VI.7 – Traitement des données (IV). Reconstruction d'un ellipsoïde (a) et d'une superquadrique (b), et affichage de leurs projections.

VI.3.2 Modèles harmoniques

VI.3.2.1 Cas 2D

Nous commençons par traiter le cas 2D, car il permet d'illustrer nos choix de façon relativement simple. Le traitement des jeux de données (I) et (II) est présenté par les figures VI.8 et VI.9, sur lesquelles sont également affichées les reconstructions elliptiques (courbes en pointillés). Nous rappelons que la reconstruction de contours harmoniques revient à minimiser le critère des moindres carrés (approché)

$$\tilde{\mathcal{K}}(\mathbf{x}) = \|\mathbf{d} - \mathcal{A}(\mathcal{P}_{\mathbf{x}})\|^2, \quad (\text{VI.6})$$

où $\mathcal{P}_{\mathbf{x}}$ est un polygone qui approche $\mathcal{C}_{\mathbf{x}}(\theta)$, lui-même défini par

$$M(\theta) = G + \rho_{\mathbf{x}}(\theta) \boldsymbol{\xi} (\cos \theta, \sin \theta)^t, \quad \theta \in (0, 2\pi), \quad (\text{VI.7})$$

où la fonction $\rho_{\mathbf{x}}$ dépend linéairement des paramètres $\mathbf{x}$ (voir l'équation (III.22) page 53), G est une position autour de laquelle le contour est étoilé, et $\boldsymbol{\xi}$ est une matrice 2×2 qui prend en compte un système d'axes d'inertie. Le polygone $\mathcal{P}_{\mathbf{x}}$ est défini par les positions $M_j = \mathcal{C}_{\mathbf{x}}(\theta_j)$, $j = 0, \dots, n_{\mathbf{x}} - 1$, où $\theta_j = 2j\pi/n_{\mathbf{x}}$. Nous définissons l'ensemble des contraintes sur $\mathbf{x}$ par

$$\forall j \in \{0, \dots, n_{\mathbf{x}} - 1\}, \quad \rho_{\mathbf{x}}(\theta_j) > \rho_{\min}, \quad (\text{VI.8})$$

où $\rho_{\min} > 0$ est un seuil arbitraire. Comme ces contraintes sont linéaires en $\mathbf{x}$, le domaine sur lequel nous minimisons $\tilde{\mathcal{K}}$ est convexe, et nous utilisons l'algorithme de gradient projeté (déjà présenté pour le modèle par voxels) pour son optimisation locale.

Avant de commenter les reconstructions harmoniques de l'objet (I), nous abordons le jeu de données (II) afin d'illustrer les difficultés liées à la méthode, et notamment le choix des contraintes sur les paramètres $\mathbf{x}$. Il s'agit d'un cas complexe au sens où le contour réel n'est pas étoilé par rapport à son centre de gravité. Nous vérifions clairement sur la figure VI.8 (a,b) que des estimations au préalable du centre de gravité de l'objet et de ses axes d'inertie, suivies de l'estimation de $\mathbf{x}$, ne conduisent pas à d'excellents résultats. Dans cette situation, nous montrons que l'inclusion des coordonnées de G dans l'ensemble des paramètres $\mathbf{x}$ fournit des résultats plus satisfaisants (cf. graphe (c)). Néanmoins, nous ne procédons pas à cette inclusion dans le cas général de façon à limiter le nombre de paramètres. Par ailleurs, ces simulations illustrent de façon très claire la nécessité d'introduire la contrainte (VI.8) avec un paramètre $\rho_{\min} > 0$. En effet, le choix $\rho_{\min} = 0$ mène à la formation de zones singulières (« points d'accumulation », cf. graphes (b,c)) que nous souhaitons éviter, alors que le choix $\rho_{\min} > 0$ favorise la régularité des contours (graphe (d)).

Enfin, les reconstructions de l'objet (I) par la méthode harmonique (figure VI.9) s'avèrent de très bonne qualité pour des valeurs raisonnables de L ($L \geq 4$). Nous remarquons que les composantes basses fréquences de ces reconstructions (notamment pour les reconstructions correspondant à $L = 2$ et $L = 3$) sont en général bien distinctes. Cette remarque mène au choix de l'estimation *simultanée* des paramètres $\mathbf{x}$, en excluant une procédure d'estimation successive des basses puis des hautes fréquences de $\mathcal{C}^*$, comme nous l'avons envisagé au chapitre III.

VI.3.2.2 Cas 3D

Les reconstructions des objets (III) et (IV) par les méthodes harmoniques sont présentées sur les figures VI.10 et VI.11. Le développement harmonique choisi n'est pas tout à fait celui que nous avons présenté au chapitre III (§ III.4.2). Pour simplifier la mise en œuvre, nous avons opté pour la décomposition de $\rho(\boldsymbol{\theta})$ sur la famille formée du produit tensoriel des familles $\{\cos l\theta, \sin l\theta, l \leq L\}$ et $\{\cos k\varphi, \sin k\varphi, k \leq K\}$, ce qui n'altère pas fondamentalement le modèle (III.24). Nous montrons que ces développements mènent à des formes variées, même pour les valeurs faibles de K et L . En pratique, nous limitons ces valeurs ($K \leq 3, L \leq 2$) de façon à majorer le nombre n de paramètres du contour ($n \leq 25$).

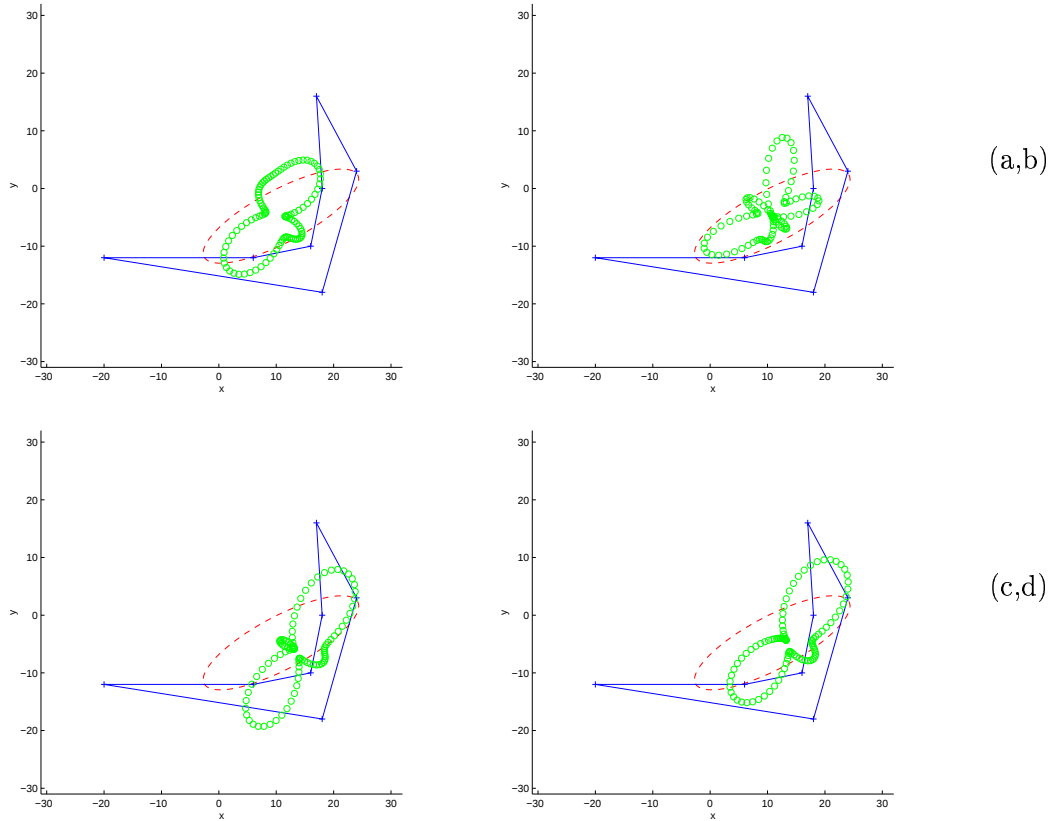


Figure VI.8 – Reconstructions de l'objet (II) par la méthode harmonique. Sont représentés : l'objet réel en traits pleins, le contour elliptique en pointillés, et le contour harmonique en points circulaires. a,b) Développements aux ordres $L = 4$ et $L = 6$, en fixant $\rho_{\min} = 0$. c) Développement à l'ordre $L = 4$, en incluant les coordonnées du centre G parmi l'ensemble des paramètres. d) Développement à l'ordre $L = 4$, avec $\rho_{\min} = 0, 1$ (pour une scène de taille 64×64).

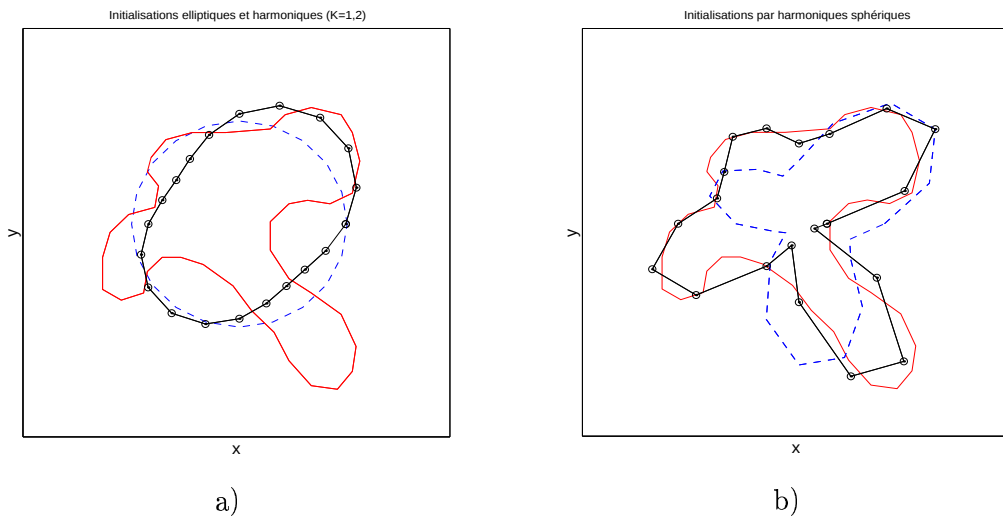


Figure VI.9 – Reconstructions elliptiques et harmoniques du contour (I), représenté en traits pleins. a) Reconstructions elliptique (en pointillés) et par développement harmonique (en points circulaires) pour $L = 2$. b) Développements aux ordres $L = 3$ (en pointillés) et $L = 4$ (en points circulaires). Les contours sont approchés par des polygones à $n_x = 20$ sommets.

Les reconstructions illustrées en figures VI.10 et VI.12 montrent que cette approche est très efficace sur le jeu de données (III), avec un très faible nombre de paramètres. En revanche, l'objet (IV) est fortement non convexe, et l'approche harmonique ne permet pas une reconstruction « optimale » en un temps raisonnable. Toutefois, la figure VI.11 montre une amélioration très nette des résultats par rapport aux reconstructions à faible nombre de paramètres (cf. figure VI.7). La difficulté de cette reconstruction est accrue par le fait que l'estimée $\hat{\xi}_3$ du premier axe principal d'inertie de $\mathcal{C}^*$ ne coïncide pas avec l'axe $\xi_3(\mathcal{C}^*)$ de « pseudo-symétrie » de $\mathcal{C}^*$ (en d'autres termes, l'ellipsoïde de la figure VI.7 n'est pas orienté suivant l'axe $\xi_3(\mathcal{C}^*)$ du « pied » du champignon). L'estimée $\hat{\xi}_3$ est primordiale pour la discrétisation de $\mathcal{C}_x$ en le polyèdre $\mathcal{P}_x$, car nous définissons les sommets de $\mathcal{P}_x$ par un ensemble de « tranches » orthogonales à cet axe (nous illustrons ce point par les représentations polyédriques de la figure VI.11, et notamment la troisième).

Enfin, nous étudions l'influence de la discrétisation de $\mathcal{C}_x$ en $\mathcal{P}_x$ sur la qualité des reconstructions. Pour cela, nous traitons le jeu de données (III) par le modèle harmonique à 14 paramètres ($K = 3, L = 1$) et une discrétisation variable du modèle. La figure VI.13 illustre les évolutions des deux critères définis au § VI.1.2 en fonction du nombre n_x de sommets. En particulier, le temps CPU évolue linéairement en fonction de n_x . Nous préconisons finalement une discrétisation relativement grossière, à moins de 100 sommets et de 250 facettes (le nombre de facettes étant de l'ordre de $2n_x$).

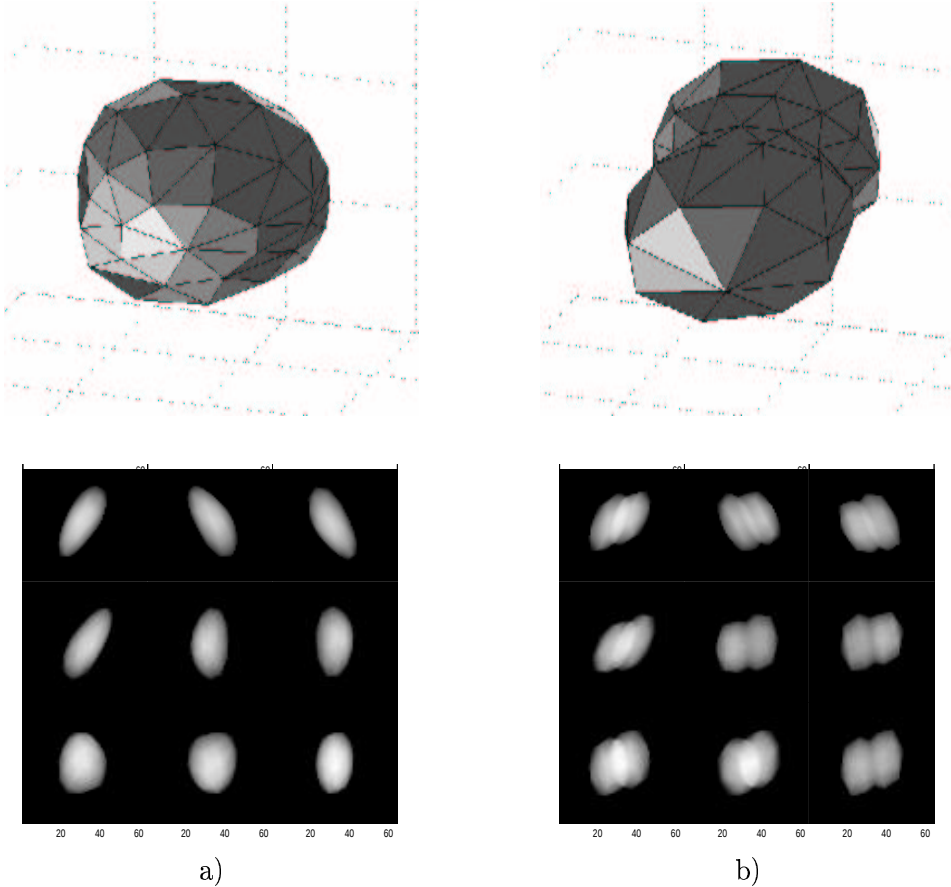


Figure VI.10 – Traitement du jeu de données (III) par la méthode harmonique. Les surfaces sont discrétisées en des polyèdres à 59 sommets et 114 facettes triangulaires. Les développements harmoniques sont tronqués aux ordres $K = 1, L = 1$ (a) et $K = 2, L = 1$ (b). Cela mène à l'estimation de $n = 8$ et $n = 11$ paramètres a_l^k .

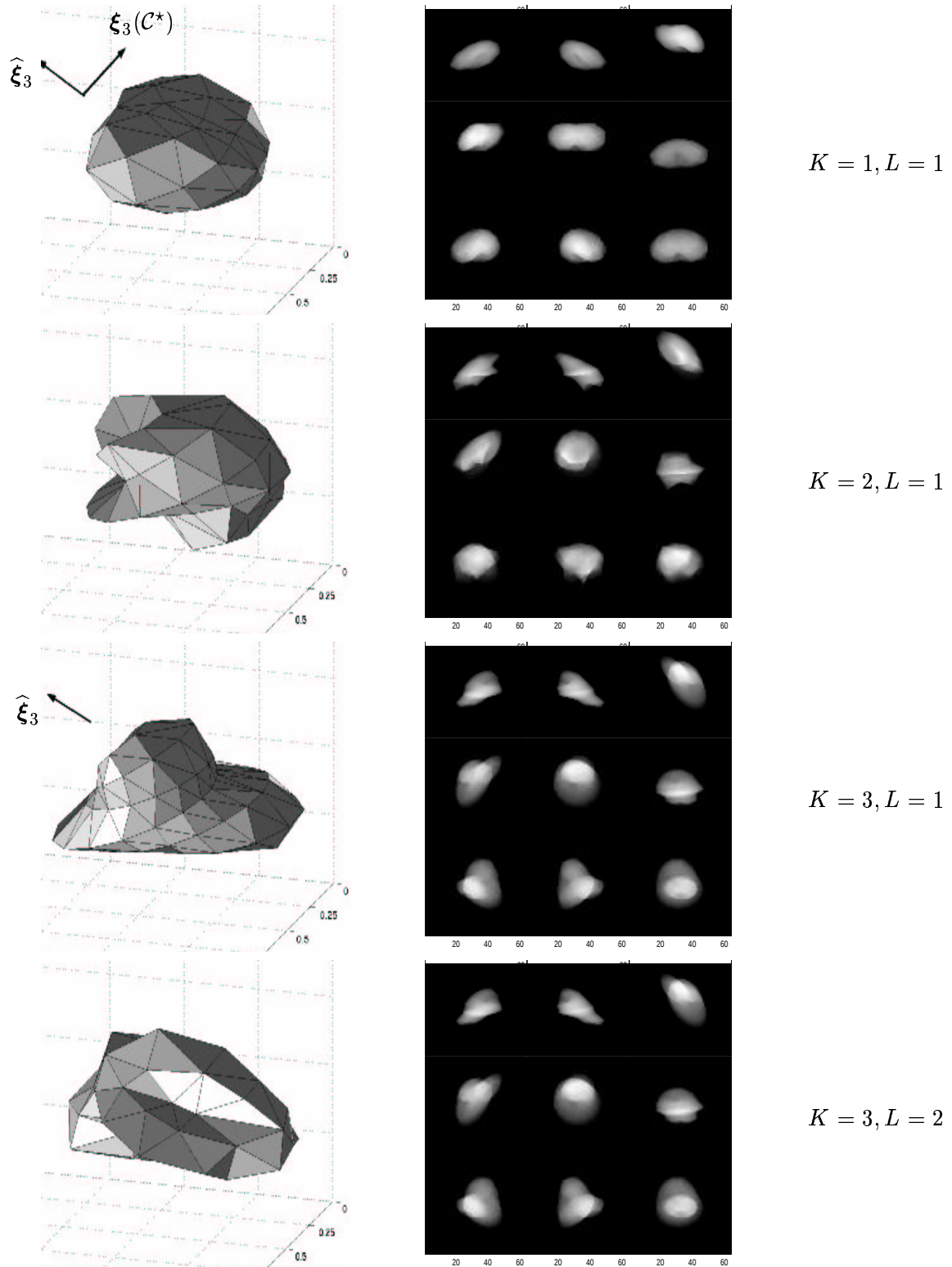


Figure VI.11 – Traitement du jeu de données (IV) par la méthode harmonique. Estimations de $n = 8, 11, 14, 25$ paramètres a_i^k respectivement. L'axe $\xi_3(\mathcal{C}^*)$ est l'axe de pseudo-symétrie de $\mathcal{C}^*$, et $\hat{\xi}_3$ est l'estimée du premier axe principal d'inertie de $\mathcal{C}^*$.

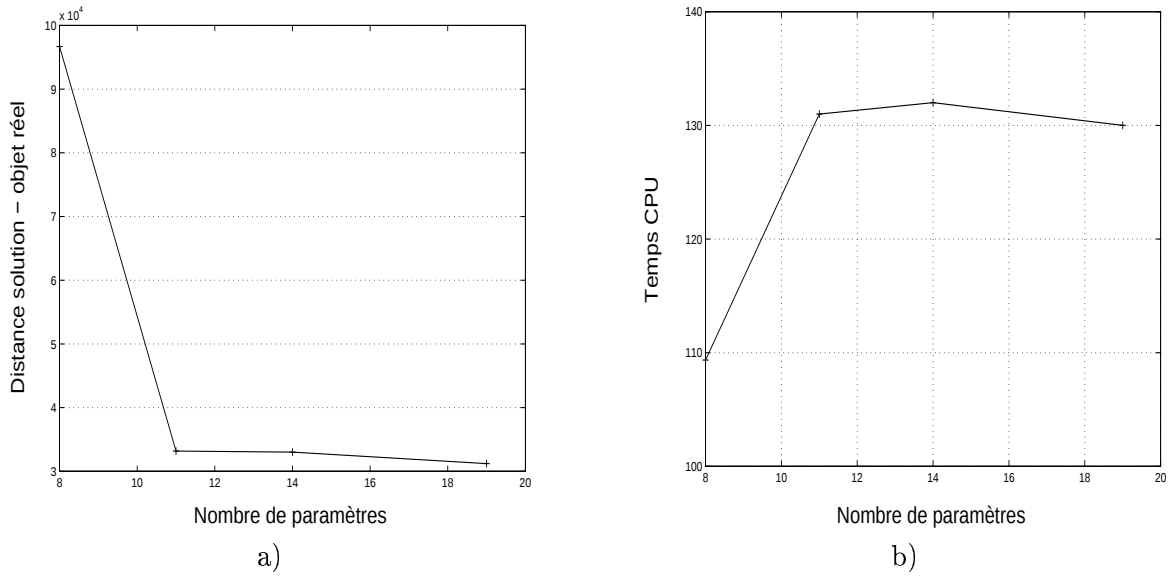


Figure VI.12 – Traitement des données (III) par la méthode harmonique. Influence du nombre n de paramètres du développement (paramètres a_l^k) sur la qualité des reconstructions, mesurée par les deux critères définis au § VI.1.2, où $n^\dagger = 128^3$ voxels. Les temps CPU obtenus pour $n = 16, 32$ et 64 sont sensiblement égaux car la convergence de l'algorithme local est plus rapide dans les deux derniers cas.

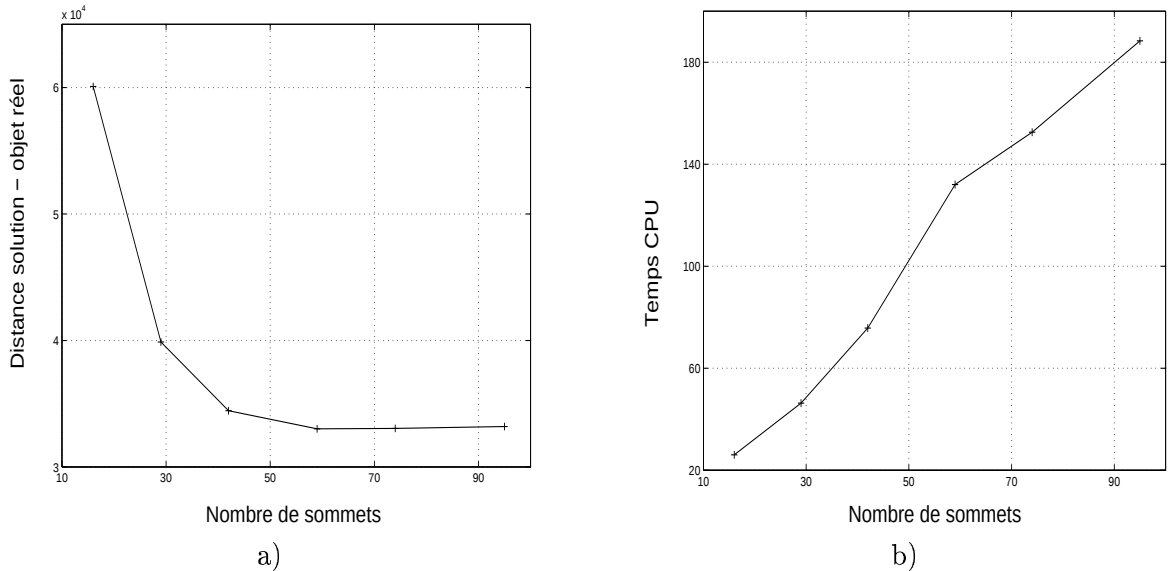


Figure VI.13 – Traitement des données (III) par la méthode harmonique ($K = 3, L = 1, n = 14$). Étude de l'influence de l'approximation du contour $\mathcal{C}_x$ (en le polyèdre $\mathcal{P}_x$) sur la reconstruction harmonique. Sont représentés les deux critères de qualité en fonction de n_x , nombre de sommets de $\mathcal{P}_x$.

VI.3.2.3 Conclusions

Nous avons montré que l'approche harmonique permet de reconstruire une grande diversité de formes en un temps convenable. Bien sûr, son but n'est pas de reconstruire de façon *exacte* le contour $\mathcal{C}^*$, mais d'extraire ses caractéristiques basse fréquence. Dans la suite, nous montrons qu'une approche locale plus sophistiquée (l'approche par splines d'ordre un) permet d'affiner ces reconstructions.

VI.4 Approche par splines d'ordre un

VI.4.1 Cas 2D

Approche par moindres carrés. Nous illustrons les performances de l'approche polygonale sur le jeu de données (I) présenté en figure VI.1. Tout d'abord, considérons la reconstruction d'un polygone $\mathbf{v} = [v_1, \dots, v_n]^t$ à n sommets, pour n fixé. L'approche la plus simple (cf. § IV.2.2) est d'estimer le polygone au sens des moindres carrés (l'estimateur est noté $\hat{\mathbf{v}}_{\text{MC}}(n)$), en minimisant le critère

$$\mathcal{K}(\mathbf{v}) = \|\mathbf{d} - \mathcal{A}(\mathbf{v})\|^2. \quad (\text{VI.9})$$

Les reconstructions correspondantes sont illustrées sur la figure VI.14 pour diverses valeurs de n . L'algorithme utilisé est de type Gauss-Seidel (GS), et le nombre d'itérations nécessaires à sa convergence dépend notamment de n . Typiquement, ce nombre varie de 30 à 200 lorsque n décrit l'intervalle $[10, 100]$ (voir la figure VI.18). L'initialisation de l'algorithme GS est la discrétisation de l'ellipse produite par la méthode des moments (voir la courbe en pointillés de la figure VI.9 (a)). Comme nous l'avons vu au chapitre IV (§ IV.3.1), cet algorithme évite la formation de contours qui s'« auto-intersectent ». Toutefois, nous remarquons que les formes produites sur la figure VI.14 sont très irrégulières (au sens de la douceur locale). Cet argument motive l'utilisation de l'approche régularisée développée au paragraphe suivant. Par ailleurs, la qualité des reconstructions en terme de distance à l'objet réel dépend fortement de n . Plus n est important, plus l'estimation de $\mathbf{v}$ est un problème mal conditionné. En outre, les reconstructions de la figure VI.14 (b) illustrent une forte dépendance du résultat à l'initialisation, ce qui traduit le grand nombre de minima locaux de $\mathcal{K}$ pour les fortes valeurs de n .

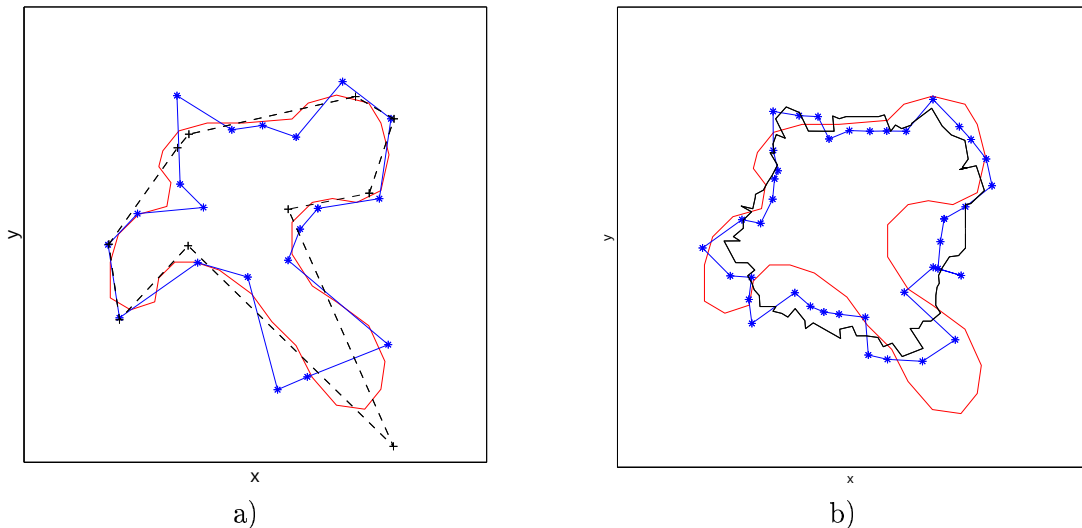


Figure VI.14 – Traitement du jeu de données (I) par l'approche polygonale; estimation $\hat{\mathbf{v}}_{\text{MC}}(n)$ par moindres carrés pour $n = 10, 20, 50, 100$ sommets. Outre l'objet réel, sont représentés : a) $\hat{\mathbf{v}}_{\text{MC}}(10)$ et $\hat{\mathbf{v}}_{\text{MC}}(20)$, en traits pointillés et pleins, b) $\hat{\mathbf{v}}_{\text{MC}}(50)$ et $\hat{\mathbf{v}}_{\text{MC}}(100)$, en traits étoilés et pleins.

Approche régularisée. Cette approche fournit des reconstructions plus douces que les précédentes. Nous avons défini au § IV.2.2 l'estimation au sens du MAP (l'estimateur est noté $\widehat{\mathbf{v}}(n)$) par la minimisation du critère

$$\mathcal{J}(\mathbf{v}) = \mathcal{K}(\mathbf{v}) + \lambda \mathcal{R}(\mathbf{v}), \quad (\text{VI.10})$$

où le terme de régularisation est défini par

$$\mathcal{R}(\mathbf{v}) = \sum_{i=1}^n \phi(1 + \cos(A_i(\mathbf{v}))), \quad (\text{VI.11})$$

et pénalise les angles aigus aux sommets de $\mathbf{v}$. Avant de commenter les résultats de cette approche, nous revenons sur le choix de la fonction ϕ , de classe $\mathcal{C}^2$ et croissante sur $\mathbb{R}_+$. Sa convexité ne confère pas de propriétés spécifiques à $\mathcal{J}$, car l'opérateur $\mathcal{K}$ est en général non convexe, et $\mathcal{J}$ l'est donc également. En revanche, l'expérience montre qu'un potentiel non convexe mène à des reconstructions très peu satisfaisantes. En d'autres termes, la non convexité de ϕ a pour effet d'augmenter le nombre de minima locaux de $\mathcal{J}$. Par conséquent, nous optons pour le potentiel quadratique $\phi(t) = t^2$.

Influence du paramètre λ . Les reconstructions de la figure VI.15 (polygones à 20 sommets) sont toujours obtenues à partir de l'ellipse discrétisée (courbe en pointillée sur le graphe (a)). L'influence du paramètre λ , qui traduit le poids des informations *a priori* par rapport à l'attache du modèle aux données, est clairement illustrée sur cette figure. Enfin, cette étude est complétée par les calculs des distances des reconstructions à l'objet réel (au sens du critère (VI.4), où $n^\uparrow = 64 \times 64$ pixels). Ces derniers montrent que les résultats sont peu sensibles aux variations de λ sur un intervalle important ($\lambda \in [50, 600]$).

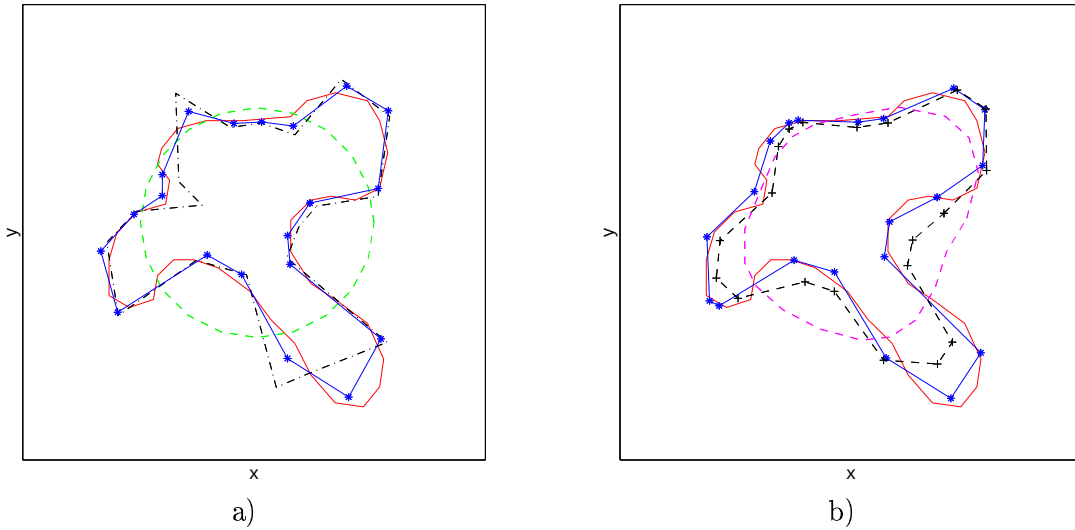
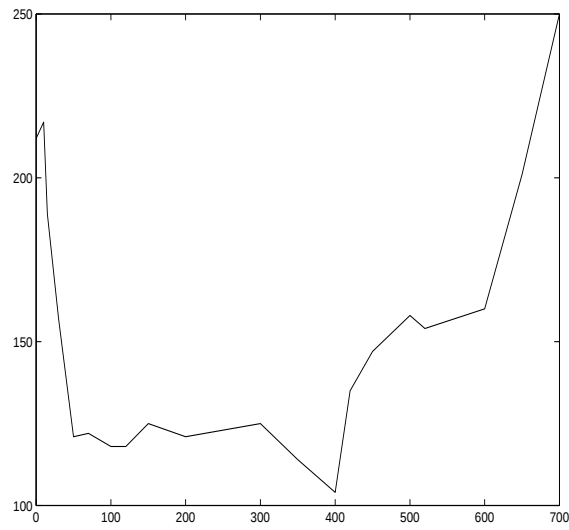
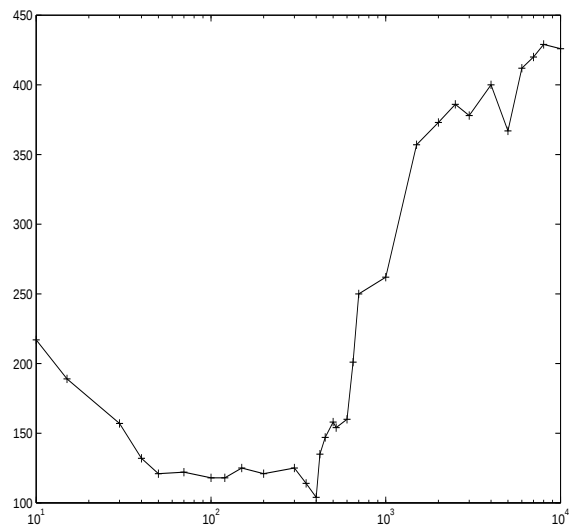


Figure VI.15 – Approche régularisée. Influence de l'hyperparamètre λ sur les reconstructions de polygones à $n = 20$ sommets. a) Solutions initiale (dérivant de la reconstruction elliptique) et finales obtenues pour $\lambda = 0$ et 100 (courbes en traits pointillés et pleins respectivement). b) Solutions obtenues pour $\lambda = 400, 700, 10000$ (courbes en traits pleins et pointillés (+, −) respectivement).



a)



b)

Figure VI.16 – Évaluation de la qualité de la reconstruction (au sens du critère (VI.4)) de polygones à 20 sommets. Ces courbes représentent la quantité $\mathcal{H}(\mathbf{f}^\uparrow, \hat{\mathbf{f}}^\uparrow)$ (où $n^\uparrow = 64 \times 64$ pixels) en fonction de λ , où λ est représenté aux échelles courante (a) et logarithmique (b).

Influence de l'initialisation. Nous commentons à présent la reconstruction de polygones à grand nombre de sommets, illustrée en figure VI.17. Comme pour l'estimation au sens des moindres carrés (qui revient à minimiser $\mathcal{K}$), le critère $\mathcal{J}$ possède de nombreux minima locaux dans les cas où n prend des valeurs importantes. L'estimée $\hat{\mathbf{v}}$ dépend donc fortement de l'initialisation $\mathbf{v}^{(0)}$ (l'algorithme d'optimisation étant local). Nous illustrons ce propos sur la figure VI.17, où $n = 100$ et les reconstructions correspondent à des initialisations de formes elliptiques (E) et harmoniques (H). Le graphe (E) montre que pour l'initialisation elliptique, les solutions obtenues sont très éloignées de $\mathcal{C}^*$, indépendamment du paramètre λ . Les polygones du graphe (H) sont les reconstructions associées à des initialisations harmoniques (voir la figure VI.9), plus proches du contour réel. Pour ces reconstructions, nous avons fixé les valeurs de λ de façon empirique, et nous obtenons des erreurs respectives de 372, 412, 209 et 125 (au sens du critère (VI.4)) pour les reconstructions optimales (relativement à λ).

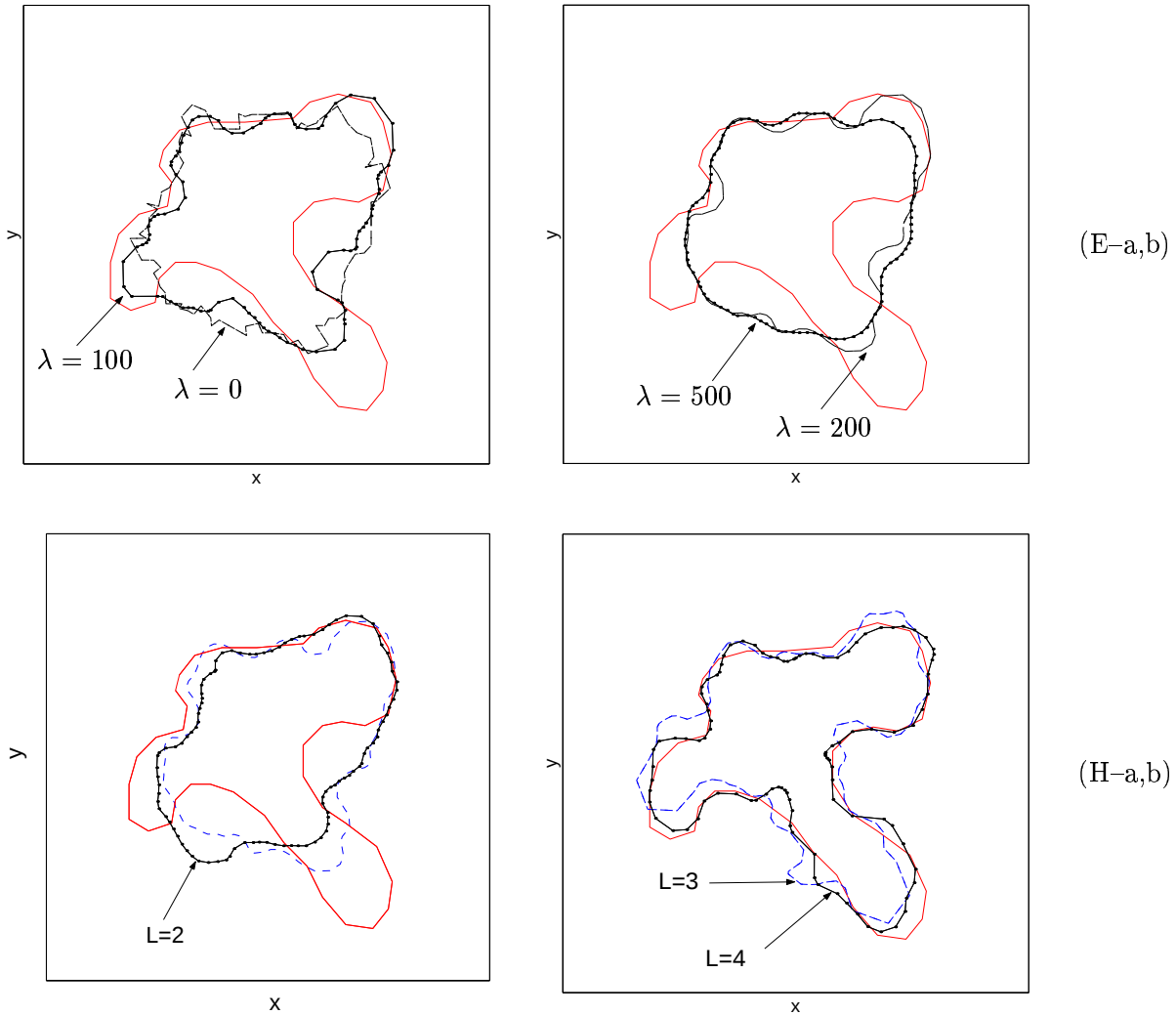


Figure VI.17 – Influence de l'initialisation sur la reconstruction pour un grand nombre de sommets ($n = 100$). (E) Initialisation par l'ellipse représentée en figure VI.9 (a). Reconstructions associées à $\lambda = 0$ et 100 (a) puis $\lambda = 200$ et 500 (b). (H) Reconstructions optimales (par rapport aux valeurs de λ) associées à des initialisations elliptiques (a, courbe en pointillés) et harmoniques (a,b). Ces initialisations sont les contours présentés sur la figure VI.9 (correspondant à $L = 2, 3, 4$). Les valeurs respectives de λ sont obtenues empiriquement : $\lambda = 200, 100, 75, 75$.

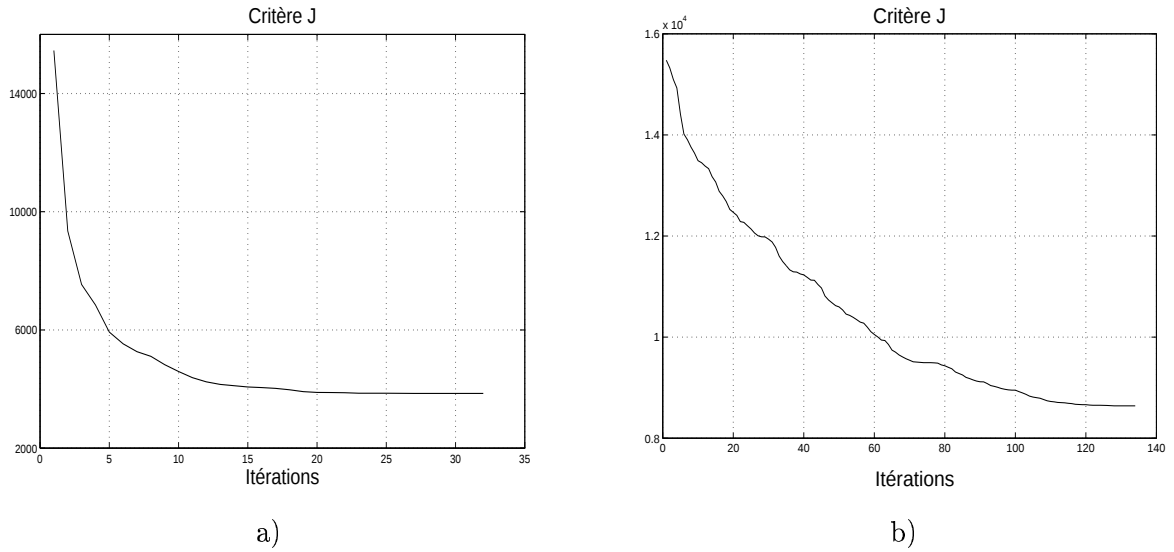


Figure VI.18 – Convergence de l'algorithme GS pour la minimisation du critère pénalisé (VI.10). Évolution du critère $\mathcal{J}(\mathbf{v}^{(k)})$ pour $n = 20$ sommets et $\lambda = 100$ (a), pour $n = 100$ sommets et $\lambda = 200$ (b). Les reconstructions sont par ailleurs illustrées sur les figures VI.15 (a) et VI.17 (E-b).

VI.4.2 Cas 3D

Nous illustrons à présent l'approche polyédrique à l'aide des jeux de données (III) et (IV). Comme dans le cas 2D, nous utilisons les méthodes par contours paramétriques « globaux » pour fournir les reconstructions initiales. Avant de présenter les résultats, nous revenons sur le choix de l'algorithme d'optimisation locale du critère $\mathcal{J}(\mathbf{V})$. Nous montrons sur la figure VI.19 qu'un algorithme sommet par sommet de type Gauss-Seidel (GS) est nettement plus lent qu'un algorithme de descente globale qui modifie l'ensemble des sommets simultanément (voir également la discussion de la partie V.3 du chapitre V). De plus, nous montrons que la charge calculatoire de la procédure GS s'accroît nettement pour les modèles à grand nombre de sommets.

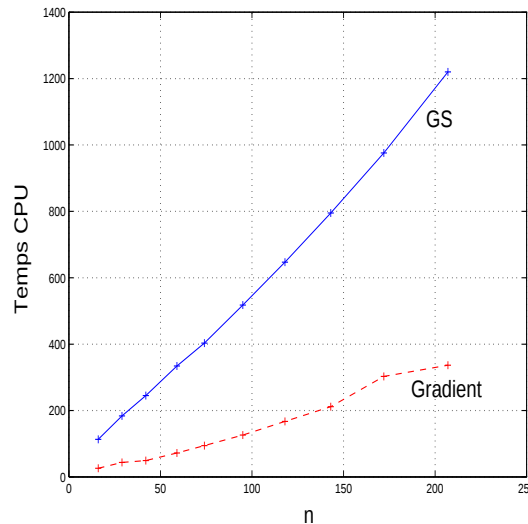


Figure VI.19 – Choix de l'algorithme d'optimisation locale. Évaluation en fonction du nombre n de sommets des coûts numériques des algorithmes de descente sur $\mathbf{V}$ (gradient) et de descente sommet par sommet (GS). Ces évaluations sont faites à partir des données (IV), et correspondent à $K = 10$ itérations des algorithmes.

Pour le jeu de données (III), nous avons vu que les reconstructions harmoniques à plus de 11 paramètres sont de bonne qualité (figure VI.12). L'approche polyédrique permet dans ce cas d'affiner les reconstructions. Nous montrons en figure VI.20 que pour un faible nombre de sommets ($n \leq 60$), l'estimation par moindres carrés ($\lambda = 0$) s'avère suffisante et ne provoque pas d'irrégularités spatiales. Le cas des données (IV) est plus complexe au sens où l'objet $\mathcal{C}^*$ est fortement non convexe, et donc mal approché par les modèles paramétriques à faible nombre de paramètres. Utilisées comme initialisations de la méthode polyédrique ($\mathcal{C}^{(0)}$), ces dernières formes ne mènent pas à des reconstructions de bonne qualité, car $\mathcal{C}^{(0)}$ n'appartient pas au bassin d'attraction du minimiseur global du critère $\mathcal{J}$. Nous vérifions ce propos sur les figures VI.21 et VI.23, où les reconstructions finales ne décrivent pas le « pied » du « champignon ». Par contre, le choix de formes harmoniques pour $\mathcal{C}^{(0)}$ est davantage approprié (voir la figure VI.24).

Pour chacune des initialisations, nous montrons l'influence du terme de régularisation par le biais du paramètre λ . De très faibles valeurs de λ mènent souvent à des formes irrégulières (voir en particulier la première reconstruction de la figure VI.25, qui possède de multiples auto-intersections), alors que des valeurs importantes ($\lambda \geq 5$) mènent à des surfaces de nature ellipsoïdale. La qualité de l'ensemble des reconstructions (pour des valeurs moyennes de λ ($\lambda \approx 1$)) est finalement illustrée par le tableau VI.2. Nous remarquons que les reconstructions associées à l'initialisation par superquadrique sont aussi « bonnes » (au sens de $\mathcal{H}(\hat{\mathcal{C}}, \mathcal{C}^*)$) que celles qui dérivent d'initialisations harmoniques, malgré le fait que le pied du champignon ne soit pas reconstruit. Cela est dû au faible volume occupé par cette partie sur les reconstructions de la figure VI.24.

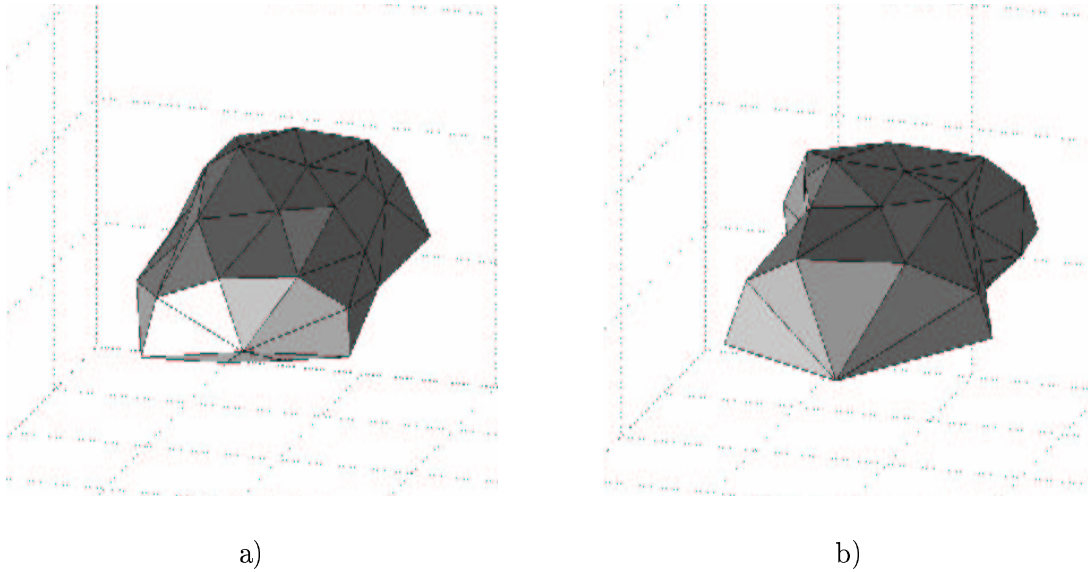


Figure VI.20 – Traitement des données (III). Reconstructions polyédriques à partir d'un ellipsoïde (a) et de la surface harmonique reconstruite en figure VI.10 (a) ($K = 1, L = 1$). Les polyèdres possèdent $n = 42$ sommets et 80 triangles, et les positions de leurs sommets sont estimées par moindres carrés, pour 10 itérations de l'algorithme du gradient. Les erreurs respectives des reconstructions sont de 22121 et 18832 voxels (où $n^\uparrow = 128^3$ voxels).

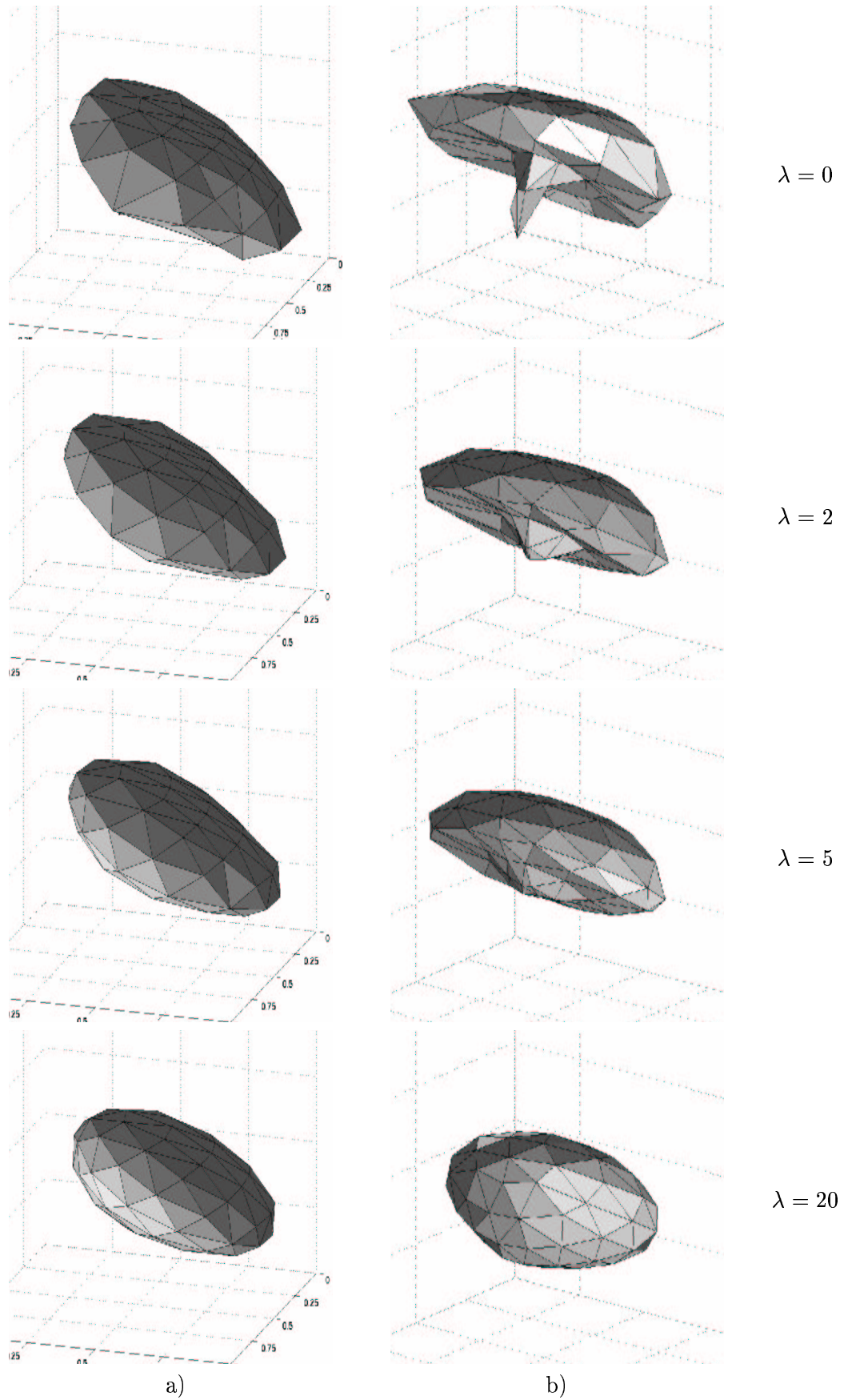


Figure VI.21 – Traitement des données (IV) par l'approche polyédrique, avec l'initialisation ellipsoïdale (objet de la figure VI.7 (a)), pour 20 itérations de l'algorithme d'optimisation locale : évolution de la reconstruction à $n = 74$ sommets et 144 triangles pour différentes valeurs du paramètre λ . Les visualisations (a) et (b) sont faites suivant deux directions orthogonales (la première est celle que nous avons adoptée jusqu'à présent).

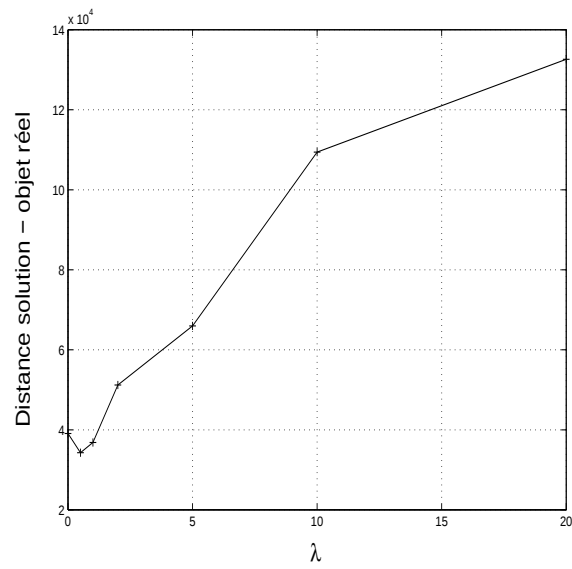
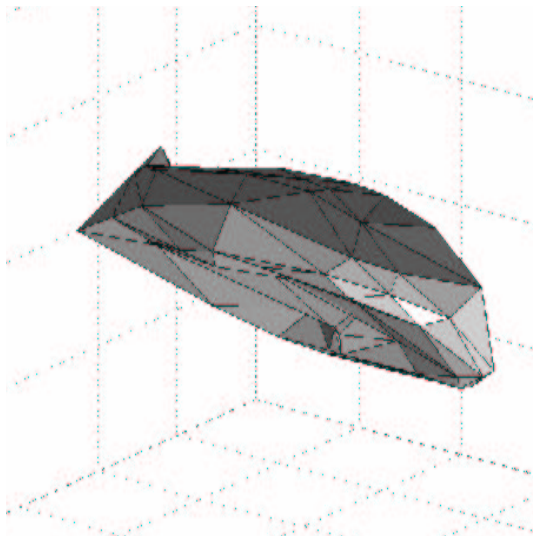
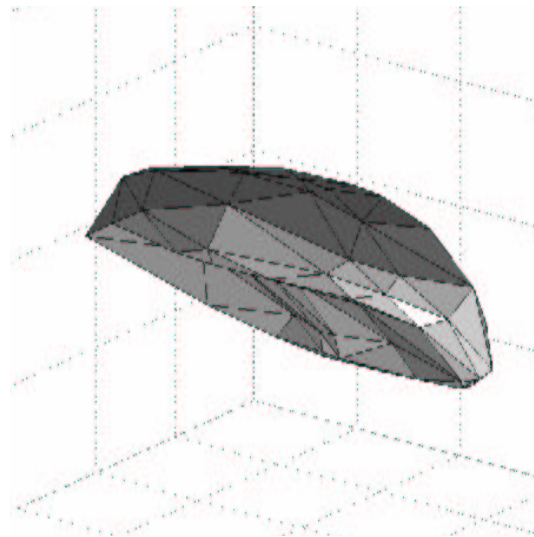


Figure VI.22 – Traitement des données (IV). Influence de l’hyperparamètre λ sur les reconstructions de polyèdres à $n = 74$ sommets et 144 triangles, où l’initialisation est ellipsoïdale (voir la figure VI.21).



$\lambda = 0,5$



$\lambda = 1$

Figure VI.23 – Traitement des données (IV) par l’approche polyédrique, avec une superquadrique pour initialisation (figure VI.7 (b)). Les polyèdres possèdent $n = 74$ sommets et 144 triangles, et λ vaut 0,5 et 1 respectivement.

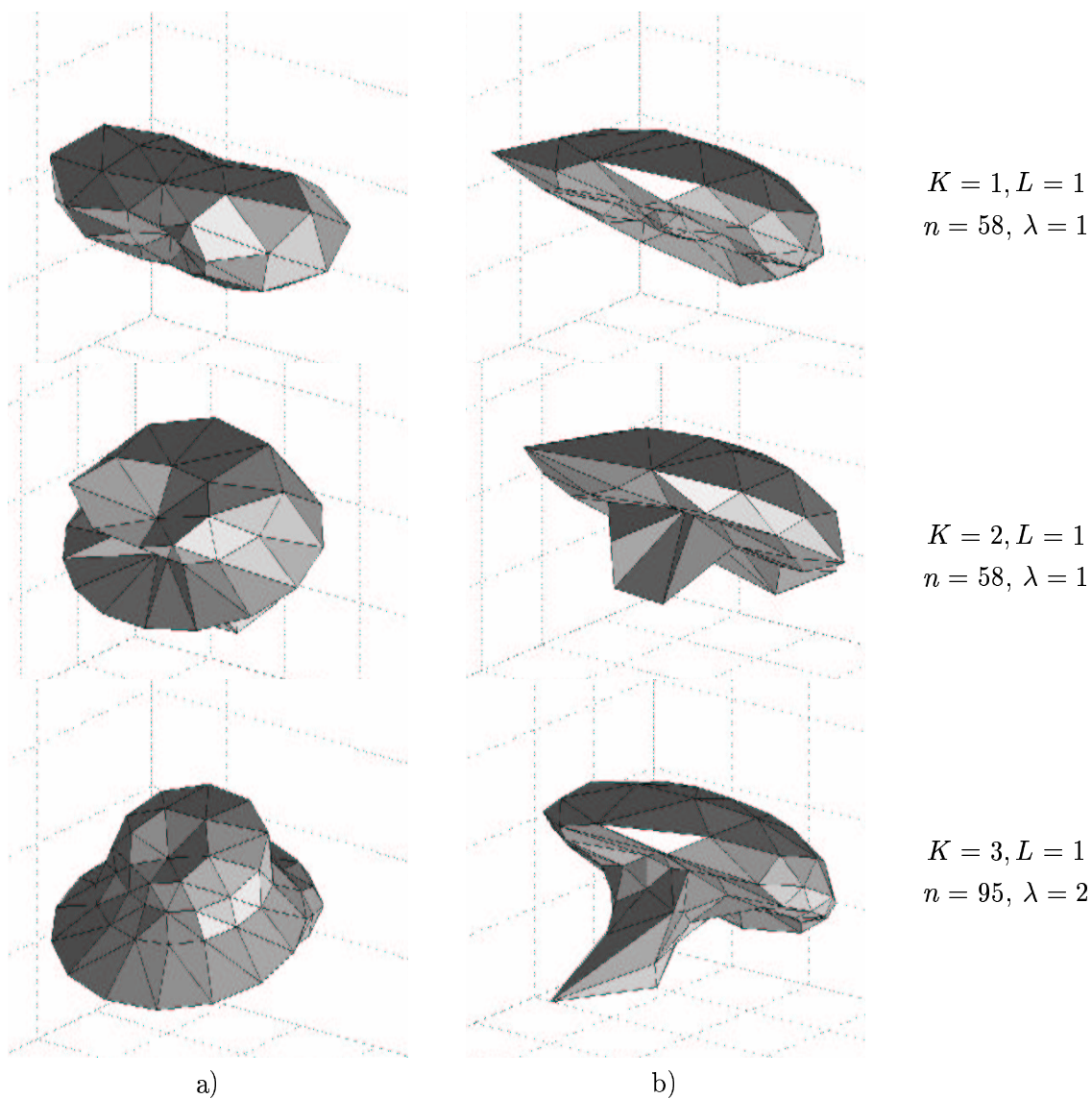


Figure VI.24 – Reconstructions polyédriques associées à des initialisations harmoniques. a) Initialisations (surfaces représentées sur la figure VI.11, mais visualisées suivant une direction orthogonale), b) Reconstructions polyédriques associées.

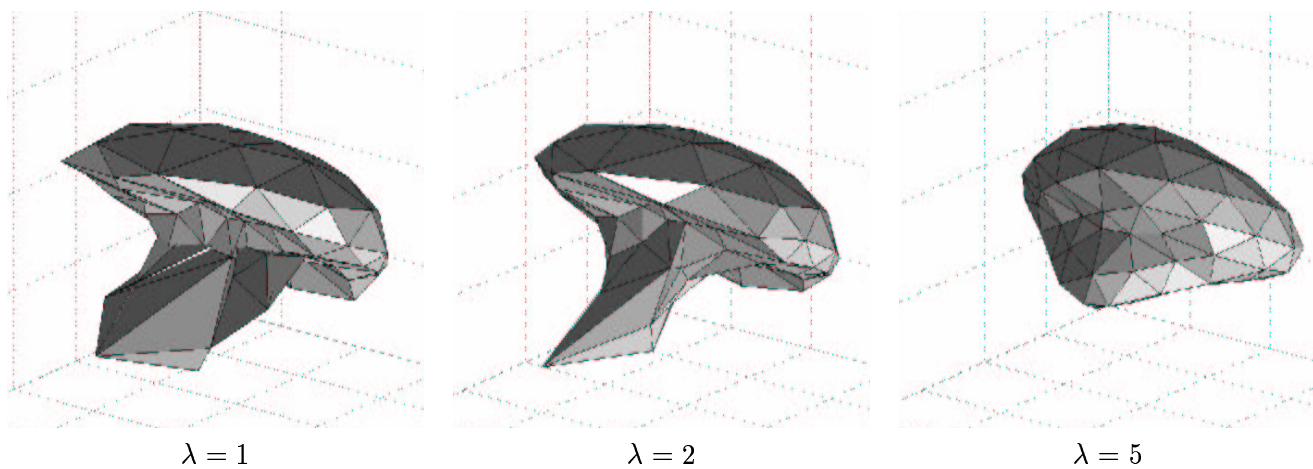


Figure VI.25 – Reconstructions polyédriques à $n = 95$ sommets obtenues à partir de l'initialisation harmonique présentée en figure VI.11 ($K = 3, L = 1$), et de différentes valeurs de λ .

Initialisation	Ellipsoïde	Superquadrique	$n = 8$	$n = 11$	$n = 14$	$n = 14$
λ	1	1	1	1	2	5
$\mathcal{H}(\mathcal{C}^{(0)}, \mathcal{C}^*)$	$3,1 \cdot 10^5$	$1,5 \cdot 10^5$	$1,7 \cdot 10^5$	$1,8 \cdot 10^5$	$1,8 \cdot 10^5$	$1,8 \cdot 10^5$
$\mathcal{H}(\widehat{\mathcal{C}}, \mathcal{C}^*)$	$3,7 \cdot 10^4$	$4,5 \cdot 10^4$	$3,8 \cdot 10^4$	$4,4 \cdot 10^4$	$4,1 \cdot 10^4$	$1,1 \cdot 10^5$

Tableau VI.2 – Traitement du jeu de données (IV) par la méthode polyédrique. Sont indiqués la qualité des initialisations $\mathcal{C}^{(0)}$ et des reconstructions $\widehat{\mathcal{C}}$, au sens du critère $\mathcal{H}(\mathcal{C}, \mathcal{C}^*)$. Les quatre dernières colonnes correspondent aux initialisations harmoniques décrites en figure VI.11 (ce sont les trois premières surfaces de cette figure).

VI.5 Conclusions

Ce chapitre a permis d'illustrer sur des jeux communs de données synthétiques l'ensemble des méthodes présentées dans ce manuscrit. Dans un premier temps, nous avons caractérisé la méthode de reconstruction par voxels en insistant sur ses limites. Nous avons montré que la stratégie multirésolution permet de réduire singulièrement son coût numérique ainsi que l'espace mémoire nécessaire à la représentation de la scène, tout en fournissant des reconstructions de bonne qualité.

Par la suite, nous avons montré que les méthodes par contours réalisent un bon compromis entre l'encombrement mémoire inhérent au stockage des paramètres et la rapidité de la reconstruction. Outre les modèles paramétriques à très faible nombre de paramètres, les modélisations harmoniques s'avèrent appropriées pour reconstruire des surfaces fermées « basse fréquence ». D'une part, nous pouvons définir des conditions très simples sur leur paramètres pour garantir leur caractère étoilé. D'autre part, les modèles harmoniques à faible nombre de paramètres induisent une grande variété de formes.

Enfin, nous préconisons les modélisations par splines d'ordre un pour la reconstruction de formes plus complexes. Du fait de la multimodalité de l'énergie *a posteriori*, la reconstruction est relativement sensible à l'initialisation. En fait, cette sensibilité est accrue lorsque le nombre de sommets est important ($n > 100$). Nous sélectionnons cette solution initiale à partir d'un modèle harmonique à faible nombre de paramètres (inférieur à 20), ce qui correspond à un développement en série de Fourier tronqué à un ordre L inférieur à 3 (l'initialisation en elle-même constitue donc une reconstruction de surface harmonique). Une alternative à cette stratégie est d'utiliser une technique multirésolution, qui consiste à :

- augmenter progressivement le nombre de sommets du modèle ;
- utiliser une reconstruction à faible nombre de sommets (par exemple $n/2$) pour former la reconstruction initiale $\mathcal{P}^{(0)}$ à n sommets.

Nous avons déjà montré l'intérêt d'une telle stratégie (Soussen et Mohammad-Djafari, 1999a,b), et nous comptons la développer davantage dans le futur.

CONCLUSIONS

Le but de ce travail a été d'étudier et de concevoir des méthodes adaptées pour la reconstruction tomographique d'un objet tridimensionnel compact et homogène dans un fond homogène, dans des contextes où les données sont limitées en nombre et en angles. Nos contributions sont à la fois d'ordre méthodologique et algorithmique.

Dans un premier temps, nous avons réalisé une synthèse sur les méthodes par voxels, qui discrétisent le volume d'intérêt en un ensemble fini de voxels parallélépipédiques, puis estiment leur densités. Sous leur forme la plus simple, ces méthodes reconstruisent des scènes homogènes par morceaux. Nous avons montré leurs limitations, puis nous avons proposé une stratégie multirésolution afin de réduire la taille de la paramétrisation et par là même la charge calculatoire.

Par la suite, nous avons étudié les méthodes qui modélisent explicitement la surface fermée de l'objet, puis la reconstruisent directement à partir des données radiographiques. Nous avons opté pour des modèles paramétriques, dont le choix dépend clairement du degré d'imperfection de la surface à reconstruire. Par exemple, la reconstruction de surfaces qui ne comportent que des parties « basse fréquence » peut être menée de façon efficace par des méthodes simples, qui reposent sur des modèles à faible nombre de paramètres (modèles de type harmoniques). En revanche, la reconstruction de surfaces complexes (comportant des inhomogénéités spatiales de nature inconnue) impose des modèles plus sophistiqués. Nous avons montré l'intérêt des modèles par splines, et avons conçu la reconstruction d'une forme polyédrique (spline d'ordre un) en nous plaçant dans le contexte bayésien, et en incorporant des informations *a priori* sur la douceur de la surface. La reconstruction consiste à estimer les positions des sommets V_i du polyèdre à partir des données, en minimisant l'énergie *a posteriori* $\mathcal{J}(V_1, \dots, V_n)$. L'originalité de ce travail concerne essentiellement le choix de la régularisation, et la mise en œuvre de l'optimisation de $\mathcal{J}$. La régularisation porte sur les angles solides en les sommets V_i , et la minimisation est menée par des algorithmes locaux. Nous avons envisagé plusieurs techniques, celles qui estiment simultanément l'ensemble des sommets (méthode itérative de descente) et celles qui ré-estiment un à un chacun des sommets V_i (méthode itérative de descente par blocs, où le i -ème bloc est associé à V_i). Contrairement au cas 2D, nous préférons les premières dans le cas 3D, car elles s'avèrent nettement plus rapides. Dans les cas 2D et 3D, les contributions techniques que nécessitent ces deux algorithmes concernent notamment la manipulation du modèle polyédrique, l'étude détaillée de ses projections et de ses auto-intersections.

Le chapitre de simulations a illustré les différents choix de modèles par contour, et la pertinence du modèle polyédrique pour la reconstruction de surfaces complexes. Nous avons montré les performances et les caractéristiques de cette méthode sur des jeux de données synthétiques, en dégagant notamment la sensibilité du résultat à la solution initiale, en particulier pour les modèles à grand nombre de sommets. Pour produire une solution initiale de bonne qualité, nous prôtons une surface harmonique dont les paramètres sont estimés au préalable au sens des moindres carrés. En d'autres termes, nous réalisons deux reconstructions, qui permettent d'estimer successivement les basses fréquences et les hautes fréquences du contour.

De façon générale, la méthode développée s'avère particulièrement adaptée à la reconstruction de surfaces complexes comportant des inhomogénéités spatiales. Ses limitations concernent la reconstruction de zones de *discontinuités* spatiales. En effet, l'information *a priori* que nous avons intégrée formalise la douceur locale de la surface. Nous avons effectué ce choix dans un cadre général, en l'absence d'hypothèse spécifique sur la nature de la surface. S'il ne limite pas la complexité des surfaces que nous pouvons reconstruire, il n'est bien sûr pas le plus adapté à la reconstruction de surfaces comportant des discontinuités spatiales en certains points. Pour ces dernières, d'autres modèles, prenant

en compte ces discontinuités dans leur structure, sont plus adaptés.

Globalement, nous dégageons deux principales perspectives de ces travaux :

1. La première est d'ordre applicatif. Il nous apparaît primordial d'appliquer la méthode à un jeu de données réelles. C'est une étape nécessaire à l'évaluation de ses performances. Le contexte de notre étude (projections limitées en nombre et en angles, objet homogène et compact) se prête davantage à une application en contrôle non destructif, mais il n'est pas exclu d'appliquer nos travaux dans le domaine médical.
2. La seconde perspective est d'ordre algorithmique, elle concerne la mise en œuvre de la reconstruction polyédrique. Nous pouvons généraliser la reconstruction séquentielle d'une surface harmonique et d'une surface polyédrique par le biais d'une stratégie multirésolution, où la notion de résolution correspond au nombre de sommets. Elle consiste à reconstruire successivement des polyèdres à nombre croissant de sommets, et à considérer une reconstruction à faible nombre de sommets pour produire l'initialisation à la résolution immédiatement supérieure.

ANNEXE A

MÉTHODE PAR VOXELS : CARACTÈRE LIPSCHITZ DE $\nabla \mathcal{J}$

Nous étudions dans cette annexe le caractère lipschitzien des gradients des énergies $\mathcal{J}(\mathbf{f})$ et $\mathcal{J}_\gamma(\mathbf{f}_v)$ introduites au chapitre II. Cette propriété est suffisante à la convergence de l'algorithme du gradient projeté décrit au § II.3 vers le minimiseur de la fonctionnelle d'énergie. Les énergies $\mathcal{J}$ et $\mathcal{J}_\gamma$ correspondent respectivement aux cas :

- où l'on réalise une discrétisation *complète* de la scène par n voxels. La fonction $\mathcal{J}$ est définie sur $(0, 1)^n$ ou $\mathbb{R}_+^n$.
- où l'on fixe les valeurs d'un certain nombre de voxels à 0. Nous parlons de représentation *creuse*. Le vecteur $\mathbf{f}_v \in (0, 1)^v$ correspond alors à la région $\mathcal{V} \subset \{1, \dots, n\}$.

A.1 Représentation complète

Nous reprenons ici le critère (II.11) :

$$\begin{aligned} \mathcal{J}(\mathbf{f}) &= \mathcal{K}(\mathbf{f}) + \lambda \mathcal{D}(\mathbf{f}) + \mu \mathcal{U}(\mathbf{f}) \\ &= \|\mathbf{d} - \mathbf{A}\mathbf{f}\|^2 + \lambda \sum_{i \sim j} \phi(f_i - f_j) + \mu \sum_i f_i, \end{aligned} \quad (\text{A.1})$$

utilisé pour la reconstruction d'un volume 3D défini sous la forme d'un ensemble de voxels $\mathbf{f} = [f_1, \dots, f_n]^t$. ϕ est une fonction paire, $\mathcal{C}^1$ et convexe sur $\mathbb{R}$. Nous étudions le caractère lipschitzien de $\nabla \mathcal{J}$ sur $\mathcal{X} = (0, 1)^n$ ou $\mathcal{X} = \mathbb{R}_+^n$. D'après la relation (A.1), nous avons :

$$\nabla \mathcal{J}(\mathbf{f}) = 2\mathbf{A}^t(\mathbf{A}\mathbf{f} - \mathbf{d}) + \lambda \left[\dots, \sum_{j \sim i} \phi'(f_i - f_j), \dots \right]^t + \mu \mathbf{e}_n, \quad (\text{A.2})$$

où $\mathbf{e}_n$ est le vecteur de taille n dont tous les éléments sont égaux à 1, et chaque somme concerne les 6 voisins du voxel i . Il est donc clair que $\nabla \mathcal{J}$ est lipschitzienne *si et seulement si* chacune des fonctions $\Phi_i : \mathcal{X} \rightarrow \mathbb{R}$ est lipschitzienne :

$$\Phi_i(\mathbf{f}) = \sum_{j \sim i} \phi'(f_i - f_j). \quad (\text{A.3})$$

Proposition 9 Une condition suffisante au caractère lipschitzien de $\nabla \mathcal{J}$ est que ϕ' soit lipschitzienne sur $\mathbb{R}$.

Preuve 9 En effet, si ϕ' est lipschitzienne de facteur K , nous avons, pour tout $\mathbf{f}$ et $\mathbf{g} \in \mathcal{X}$,

$$\begin{aligned} |\Phi_i(\mathbf{f}) - \Phi_i(\mathbf{g})| &\leq \sum_{j \sim i} |\phi'(f_i - f_j) - \phi'(g_i - g_j)| \\ &\leq K \sum_{j \sim i} |f_i - g_i - (f_j - g_j)| \\ &\leq K \left(6|f_i - g_i| + \sum_{j \sim i} |f_j - g_j| \right), \end{aligned}$$

pour un champ à 6 voisins. Par conséquent,

$$\begin{aligned} \|\nabla \mathcal{J}(\mathbf{f}) - \nabla \mathcal{J}(\mathbf{g})\|_1 &\leq 2 \|\mathbf{A}^t \mathbf{A}\|_1 \|\mathbf{f} - \mathbf{g}\|_1 + \lambda K \left[6 \|\mathbf{f} - \mathbf{g}\|_1 + \sum_i \sum_{j \sim i} |f_j - g_j| \right] \\ &\leq 2 \left[\|\mathbf{A}^t \mathbf{A}\|_1 + 6\lambda K \right] \|\mathbf{f} - \mathbf{g}\|_1. \end{aligned} \quad (\text{A.4})$$

□

Parmi les fonctions ϕ que nous avons citées au § II.2.1, nous examinons le cas :

- des fonctions $\mathcal{L}_\gamma$.
- des fonctions $\mathcal{L}_2$ – $\mathcal{L}_1$.

Ces dernières sont les seules dont la dérivée est lipschitzienne sur $\mathbb{R}$. En effet, la fonction $\phi(t) = |t|^\gamma$, $1 < \gamma < 2$ est dérivable sur $\mathbb{R}^*$ de dérivée $\phi'(t) = \gamma \operatorname{sgn}(t) |t|^{\gamma-1}$. Comme $\phi''(t) \rightarrow \pm\infty$ quand $t \rightarrow 0$, ϕ' n'a pas un caractère lipschitzien au voisinage de 0. En revanche, les fonctions ϕ de classe $\mathcal{L}_2$ – $\mathcal{L}_1$ possèdent une dérivée finie en 0. Comme, de plus, leur dérivée ϕ' est continue et bornée sur $\mathbb{R}$, elle est forcément lipschitzienne. Ce résultat est par exemple vérifié pour la fonction $\phi_T(t) = \sqrt{t^2 + T^2} - T$, ou la fonction de Huber définie en (II.7), dont la dérivée est une fonction impaire, continue, bornée et affine par morceaux :

$$\phi'_T(t) = \begin{cases} 2t/T^2 & \text{si } |t| < T, \\ 2 \operatorname{sgn}(t)/T & \text{sinon.} \end{cases} \quad (\text{A.5})$$

Cette fonction est donc bien lipschitzienne sur $\mathbb{R}$ de facteur $K = 2/T^2$.

Pour cette fonction, qui est celle que nous avons choisie au chapitre II, nous utilisons l'équivalence des normes $\|\cdot\|_1$ et $\|\cdot\|_2$ (*i.e.*, norme euclidienne $\|\cdot\|$ de $\mathbb{R}^n$) :

$$\forall \mathbf{f} \in \mathbb{R}^n, \quad \|\mathbf{f}\|_2 \leq \|\mathbf{f}\|_1 \leq \sqrt{n} \|\mathbf{f}\|_2, \quad (\text{A.6})$$

pour déduire de (A.4) une valeur du facteur L de Lipschitz de $\nabla \mathcal{J}$:

$$\begin{aligned} L &= 2\sqrt{n} \left[\|\mathbf{A}^t \mathbf{A}\|_1 + 6\lambda K \right] \\ &= 2\sqrt{n} \left[\|\mathbf{A}^t \mathbf{A}\|_1 + 12\lambda/T^2 \right]. \end{aligned} \quad (\text{A.7})$$

A.2 Représentation creuse

Dans ce cas, la structure (II.24) de la fonctionnelle $\mathcal{J}_\mathcal{V}(\mathbf{f}_\mathcal{V})$ est analogue à celle du critère $\mathcal{J}(\mathbf{f})$, défini en (A.1). La seule différence réside dans la formulation (II.25) de $\mathcal{D}_\mathcal{V}(\mathbf{f}_\mathcal{V})$, qui fait intervenir un terme additionnel séparable en les valeurs des voxels de $\mathbf{f}_\mathcal{V}$.

$$\mathcal{D}_\mathcal{V}(\mathbf{f}_\mathcal{V}) = \sum_{\substack{i,j \in \mathcal{V} \\ i \sim j}} \phi_T(f_i - f_j) + \sum_{i \in \partial \mathcal{V}} \phi_T(f_i) n_\mathcal{V}(i). \quad (\text{A.8})$$

Ainsi, l'expression de $\nabla \mathcal{J}_\mathcal{V}$ est :

$$\begin{aligned} \nabla \mathcal{J}_\mathcal{V}(\mathbf{f}_\mathcal{V}) &= 2[\mathbf{A}_\mathcal{V}]^t (\mathbf{A}_\mathcal{V} \mathbf{f}_\mathcal{V} - \mathbf{d}) + \lambda \left[\dots, \sum_{\substack{j \in \mathcal{V} \\ j \sim i}} \phi'(f_i - f_j), \dots \right]^t + \\ &\quad \lambda \left[\dots, \phi'_T(f_i) n_\mathcal{V}(i), \dots \right]^t + \mu \mathbf{e}_\mathcal{V}, \end{aligned} \quad (\text{A.9})$$

où la définition (II.26) de $n_{\mathcal{V}}(i)$ est étendue aux voxels non frontières de $\mathcal{V}$ par :

$$n_{\mathcal{V}}(i) = 0 \text{ si } i \notin \partial\mathcal{V}. \quad (\text{A.10})$$

En supposant toujours que la fonction ϕ'_T est de Lipschitz de facteur K , nous obtenons avec un raisonnement du même type que celui de la preuve 9 les inégalités suivantes valables pour tous $\mathbf{f}_v$ et $\mathbf{g}_v \in (0,1)^v$:

$$\begin{aligned} \|\nabla \mathcal{J}_{\mathcal{V}}(\mathbf{f}_v) - \nabla \mathcal{J}_{\mathcal{V}}(\mathbf{g}_v)\|_1 &\leq 2 \|\mathbf{A}_v\|^t \mathbf{A}_v\|_1 \|\mathbf{f}_v - \mathbf{g}_v\|_1 + 2\lambda K \left[\sum_{i \in \mathcal{V}} (6 - n_{\mathcal{V}}(i)) |f_i - g_i| \right] + \\ &\quad \lambda K \left[\sum_{i \in \mathcal{V}} n_{\mathcal{V}}(i) |f_i - g_i| \right] \\ &\leq 2 \left[\|\mathbf{A}_v\|^t \mathbf{A}_v\|_1 + 6\lambda K \right] \|\mathbf{f}_v - \mathbf{g}_v\|_1, \end{aligned} \quad (\text{A.11})$$

en remarquant que le nombre de voisins de i dans $\mathcal{V}$ est $6 - n_{\mathcal{V}}(i)$, et pour tout i , $0 \leq n_{\mathcal{V}}(i) \leq 6$. Par conséquent, $\nabla \mathcal{J}_{\mathcal{V}}$ est bien lipschitzienne, et son facteur de Lipschitz s'exprime de même qu'au paragraphe précédent sous la forme :

$$L_v = 2\sqrt{v} \left[\|\mathbf{A}_v\|^t \mathbf{A}_v\|_1 + 12\lambda/T^2 \right]. \quad (\text{A.12})$$

ANNEXE B

RECOVERY OF MOMENTS OF A COMPACT HOMOGENEOUS
OBJECT FROM TOMOGRAPHIC PROJECTION DATACharles SOUSSEN, Ali MOHAMMAD-DJAFARI, and Ken SAUER ¹

Abstract: We consider the reconstruction of a three-dimensional compact object from a small number of its tomographic projection data (radiographs). One approach to solve this problem is to split it into two successive steps: estimation of the moments of the object from the data and estimation of its shape from these moments. In this paper, we consider the first of these steps. We first present the mathematical relations between the moments of the object and those of its projections and then provide necessary conditions for their inversion. We then give the appropriate algorithm to compute the object moments from the data and present numerical evaluations of the performance of the algorithm.

¹ *Submitted to Graphical Models and Image Processing.* Charles SOUSSEN and Ali MOHAMMAD-DJAFARI (email: name@lss.supelec.fr) are with the Laboratoire des Signaux et Systèmes (CNRS – SUPÉLEC – UPS). SUPÉLEC, Plateau de Moulon, 91192 Gif-sur-Yvette Cedex, France. Ken SAUER (email: sauer@nd.edu) is with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA.

List of symbols

n	order of the evaluated moments
m	total number of projections
C_i^j	binomial coefficient
$\mathcal{P}$	2D or 3D region occupied by compact object
SNR	signal to noise ratio

TWO-DIMENSIONAL TOMOGRAPHY NOTATION

$f(x, y)$	density function
θ	angle of projection
$p(t, \theta)$	parallel projection measurement and angle θ
μ_{ij}	geometric moment of $\mathcal{P}$
$\boldsymbol{\mu}^n$	vector of n -th order geometric moments of $\mathcal{P}$
$\overline{\boldsymbol{\mu}}^n$	truncation of $\boldsymbol{\mu}^n$
$h_n(\theta)$	n -th order moment of the projection at angle θ
$\mathbf{h}^n$	vector of n -th order moments of the projections
$\mathbf{h}_i^n$	n -length vector formed by deleting $h_n(\theta_i)$ from $\mathbf{h}^n$

THREE-DIMENSIONAL TOMOGRAPHY NOTATION

$f(x, y, z)$	density function
μ_{ijk}	geometric moment of $\mathcal{P}$
$\boldsymbol{\mu}_l^n$	vector formed of variables $(\mu_{i,n-i-l,l}), i = 0, \dots, n-l$
(θ, φ)	angle of projection
$\mathbf{w}$	normal vector to projection plane
$(\mathbf{u}, \mathbf{v}, \mathbf{w})$	system of orthogonal axes in projection's coordinates
$\mathbf{t}$	position of detector in projection plane
$p(\mathbf{t}, \theta, \varphi)$	projection at angle (θ, φ)
$p(\mathbf{t}, \mathbf{w})$	projection along lines parallel to $\mathbf{w}$
$\mathcal{L}_{\mathbf{t}, \mathbf{w}}$	corresponding projection line
$h_{lq}(\theta, \varphi)$	order $l + q$ moment of a projection
$\mathbf{h}_l^n$	affine transform of the vector formed by variables $h_{l,n-l}(\theta, \varphi)$

B.1 Introduction

In many nondestructive testing (NDT) applications, one may be interested in determining the 3D shape of a body from a few of its X-ray radiographs. This problem is very closely related to general image reconstruction in X-ray tomography which consists of determining a function $f(x, y, z)$ from its line projections

$$p(\mathbf{t}, \mathbf{w}) = \int_{\mathcal{L}_{\mathbf{t}, \mathbf{w}}} f(x, y, z) dl, \quad (\text{B.1})$$

where the integration is done over the line $\mathcal{L}_{\mathbf{t}, \mathbf{w}}$ oriented by $\mathbf{w}$ and passing through $\mathbf{t}$, the position of the detector.

In this paper, we consider the case where f has a constant value c_1 inside a single region $\mathcal{P}$ and another constant value c_2 outside $\mathcal{P}$:

$$f(x, y, z) = \begin{cases} c_1 & \text{if } (x, y, z) \in \mathcal{P}, \\ c_2 & \text{elsewhere.} \end{cases} \quad (\text{B.2})$$

Without loss of generality, we assume that $c_1 = 1$ and $c_2 = 0$; the image reconstruction problem is then equivalent to the determination of the shape of the interior region $\mathcal{P}$. For the sake of simplicity, this shape, *i.e.*, the boundary of $\mathcal{P}$, will also be called the object in the following.

The problem of determining $\mathcal{P}$ from a finite numbers of its projections is well known in image reconstruction. There are at least three main approaches:

- The first approach consists of the discretization of the whole space (both outside and inside the object) into voxels. One must estimate the value of f in each voxel and then decide whether this value is set to c_1 or c_2 (Soumekh, 1988; Besag, 1986, 1989; Sauer and Bouman, 1992; Bouman and Sauer, 1993; Nikolova and Mohammad-Djafari, 1996).

The advantage of such methods is that the relation between the data and the unknown parameters (voxel values) is linear. Two main drawbacks are the huge number of the parameters to estimate and the fact that the defined model does not constrain the object to be compact or to have a closed contour.

- The second approach consists of a direct estimation of the closed contour of the object from the data. Many works, derived from this approach, have been completed (Snyder et al., 1995; Battle et al., 1998; Hanson et al., 1997). These methods can also be separated into two subcategories: — those which model the contour of the object as the level set of a continuous function $\phi(x, y, z)$ and aim to estimate ϕ from the data (Amit and Manbeck, 1993; Malladi et al., 1995; Santosa, 1996); and — those which model and estimate the geometric shape of the object from the projections (Rossi and Willsky, 1984; Milanfar et al., 1996). Here, the number of unknown parameters is drastically reduced, but the relation between the data and these parameters is no longer linear.

Two subclasses of this last category of methods are:

- the methods which use simple geometric shapes with a very small number of parameters, such as spheres or ellipsoids (Rossi and Willsky, 1984); and
- those which model the shape by a more general deformable template such as a spline or a polyhedral shape with a great number of vertices (Bresler and Macovski, 1987; Milanfar et al., 1995; Hanson et al., 1997).

- The third approach consists of splitting the shape reconstruction into two successive steps (Milanfar et al., 1996):
i) the estimation of the moments of the object from the projection data; and
ii) the estimation of the shape parameters from its moments.

In this paper, we consider this last approach and study only its first step. We propose a method to estimate the geometric moments of the object from the data. To our knowledge, the estimation of the shape from the moments is still an open problem.

In the 2D case, this problem has been first addressed by Milanfar et al. (1995, 1996) who have also proposed a method solving the second step when the object is modeled by a polygon. The present work can be considered as an extension into 3D of the first step of their method. A method providing an estimation of the vertices of a polyhedral object from its moments would complete the present work.

The organization of this paper is the following: in Section B.2, we summarize the 2D case. In Section B.3, we propose a 3D extension for the case of parallel X-ray tomography. In Section B.4, we present the algorithms and numerical procedures for both 2D and 3D cases. Finally, in Section B.5, we present some simulations and numerical evaluations of the performance of the proposed algorithm.

B.2 Summary of the 2D results

The following 2D results are mainly due to Milanfar et al. (1996). Here we summarize their work in a compact way with a slightly different notation. Using a classical parallel X-ray tomography notation, the 2D line projections are

$$p(t, \theta) = \iint_{\mathbb{R}^2} f(x, y) \delta(x \cos \theta + y \sin \theta - t) \, dx \, dy, \quad (\text{B.3})$$

where $\theta \in [0, \pi[$, $t \in \mathbb{R}$ and δ is the 1D Dirac distribution. We consider a 2D compact object $\mathcal{P}$ and define its density f by

$$f(x, y) = \begin{cases} 1 & \text{if } (x, y) \in \mathcal{P}, \\ 0 & \text{elsewhere.} \end{cases} \quad (\text{B.4})$$

With this assumption, equation (B.3) becomes

$$p(t, \theta) = \iint_{\mathcal{P}} \delta(x \cos \theta + y \sin \theta - t) \, dx \, dy. \quad (\text{B.5})$$

For any value of θ , the 1D n -th order moment of $p(t, \theta)$ is

$$h_n(\theta) = \int_{\mathbb{R}} t^n p(t, \theta) \, dt, \quad n \geq 0. \quad (\text{B.6})$$

Similarly, the geometric moments of $\mathcal{P}$ are

$$\mu_{ij} = \iint_{\mathbb{R}^2} x^i y^j f(x, y) \, dx \, dy = \iint_{\mathcal{P}} x^i y^j \, dx \, dy, \quad i, j \geq 0. \quad (\text{B.7})$$

The n -th order geometric moments of $\mathcal{P}$ are the values of μ_{ij} such that $i + j = n$. Hence, there are $n + 1$ independent n -th order moments of $\mathcal{P}$. Equations (B.5) and (B.6) yield:

$$\begin{aligned} h_n(\theta) &= \iint_{\mathcal{P}} \left(\int_{\mathbb{R}} t^n \delta(x \cos \theta + y \sin \theta - t) \, dt \right) \, dx \, dy \\ &= \iint_{\mathcal{P}} (x \cos \theta + y \sin \theta)^n \, dx \, dy \end{aligned} \quad (\text{B.8})$$

$$= \sum_{j=0}^n C_n^j \sin^j \theta \cos^{n-j} \theta \mu_{n-j, j}, \quad (\text{B.9})$$

where C_n^j denotes $\binom{n}{j}$. If $\theta \neq \pi/2$, (B.9) yields

$$h_n(\theta) = \cos^n \theta \sum_{j=0}^n C_n^j \tan^j \theta \mu_{n-j,j}. \quad (\text{B.10})$$

Note that any n -th order moment of the projections depends on only the n -th order geometric moments of the object. Consequently, for a fixed n , one needs to have at least $n+1$ values of $h_n(\theta)$, that is $n+1$ distinct projections, to estimate all coefficients $\mu_{n-j,j}$. In this case, we denote $\theta_0, \dots, \theta_n$ the $n+1$ corresponding distinct angle values and define the vectors of the n -th order moments of $\mathcal{P}$ and of the n -th order moments of the line projections by

$$\boldsymbol{\mu}^n = (\mu_{n0}, \mu_{n-1,1}, \dots, \mu_{0n})^t \text{ and } \mathbf{h}^n = (h_n(\theta_0), h_n(\theta_1), \dots, h_n(\theta_n))^t. \quad (\text{B.11})$$

At this step, we distinguish 2 cases:

1. If none of the angles θ_j is equal to $\pi/2$, the n -moment vector $\boldsymbol{\mu}^n$ is a solution of the linear system

$$\mathbf{h}^n = \mathcal{D}_1^n(\boldsymbol{\theta}) \begin{pmatrix} 1 & \tan(\theta_0) & \tan^2(\theta_0) & \dots & \tan^n(\theta_0) \\ 1 & \tan(\theta_1) & \tan^2(\theta_1) & \dots & \tan^n(\theta_1) \\ \vdots & & & & \vdots \\ 1 & \tan(\theta_n) & \tan^2(\theta_n) & \dots & \tan^n(\theta_n) \end{pmatrix} \mathcal{C}_1^n \boldsymbol{\mu}^n, \quad (\text{B.12})$$

where $\boldsymbol{\theta}$ is the vector formed by the parameters $\theta_j, j = 0, \dots, n$,

$$\mathcal{D}_1^n(\boldsymbol{\theta}) = \text{diag}[\cos^n(\theta_0), \cos^n(\theta_1), \dots, \cos^n(\theta_n)] \text{ and } \mathcal{C}_1^n = \text{diag}[C_n^0, C_n^1, \dots, C_n^n]. \quad (\text{B.13})$$

This linear system is invertible since the corresponding matrix is the product of an invertible Van Der Monde matrix (all θ_j are distinct) and two diagonal matrices with non-zero coefficients.

2. Otherwise, denote i the integer such that $\theta_i = \pi/2$. Equation (B.9) yields $h_n(\theta_i) = \mu_{0n}$. Any other parameter $\theta_j, j \neq i$ cannot be equal to θ_i since it represents another projection. We denote

$$\bar{\boldsymbol{\mu}}^n = \begin{pmatrix} \mu_{n0} \\ \mu_{n-1,1} \\ \vdots \\ \mu_{1,n-1} \end{pmatrix} \text{ and } \mathbf{h}_i^n = \begin{pmatrix} h_n(\theta_0) \\ h_n(\theta_1) \\ \vdots \\ h_n(\theta_{i-1}) \\ h_n(\theta_{i+1}) \\ \vdots \\ h_n(\theta_n) \end{pmatrix} - \mu_{0n} \begin{pmatrix} \sin^n(\theta_0) \\ \sin^n(\theta_1) \\ \vdots \\ \sin^n(\theta_{i-1}) \\ \sin^n(\theta_{i+1}) \\ \vdots \\ \sin^n(\theta_n) \end{pmatrix} \quad (\text{B.14})$$

the truncated vector of the geometric moments and an affine transform of the truncated vector of the moments of the projections. Both vectors have a size equal to n . One can derive from (B.10) the linear system

$$\mathbf{h}_i^n = \mathcal{D}_{1i}^n(\boldsymbol{\theta}) \begin{pmatrix} 1 & \tan(\theta_0) & \dots & \tan^{n-1}(\theta_0) \\ 1 & \tan(\theta_1) & \dots & \tan^{n-1}(\theta_1) \\ \vdots & \vdots & & \vdots \\ 1 & \tan(\theta_{i-1}) & \dots & \tan^{n-1}(\theta_{i-1}) \\ 1 & \tan(\theta_{i+1}) & \dots & \tan^{n-1}(\theta_{i+1}) \\ \vdots & \vdots & & \vdots \\ 1 & \tan(\theta_n) & \dots & \tan^{n-1}(\theta_n) \end{pmatrix} \bar{\mathcal{C}}^n \bar{\boldsymbol{\mu}}^n, \quad (\text{B.15})$$

where $\mathcal{D}_{1i}^n(\boldsymbol{\theta}) = \text{diag}[\cos^n(\theta_0), \dots, \cos^n(\theta_{i-1}), \cos^n(\theta_{i+1}), \dots, \cos^n(\theta_n)]$ and $\bar{\mathcal{C}}^n = \text{diag}[C_n^0, C_n^1, \dots, C_n^{n-1}]$. This system is similar to (B.12). Consequently, it is also invertible.

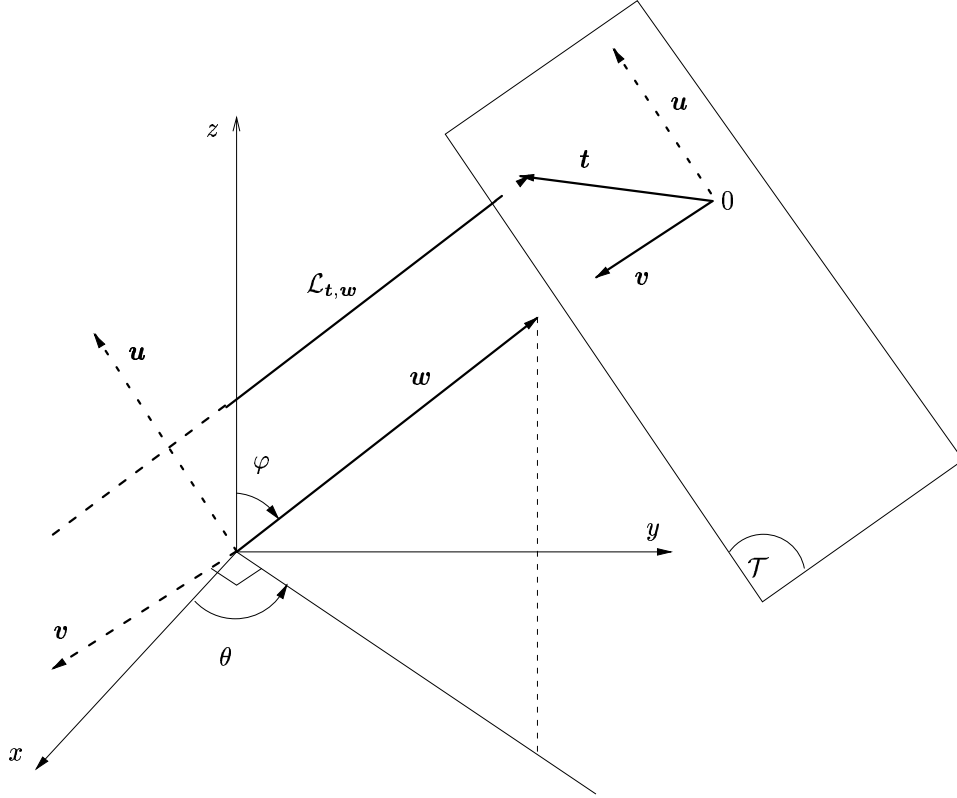


Figure B.1: Geometric definition of a projection.

It follows from this discussion that the exact reconstruction of the n -th order geometric moments of $\mathcal{P}$ is always possible, given $n + 1$ distinct projections of $\mathcal{P}$.

B.3 Reconstruction of the geometric moments in 3D case

The main contribution of this work is the extension into 3D of the previous results. To our knowledge, such work has not been done before. Our goal is to apply this extension in the field of 3D tomographic image reconstruction.

B.3.1 Basic relations

We consider here parallel projections. Denote $\mathbf{w}$ the vector directing the projection line and $(\mathcal{T})$ the plane on which the data are collected. $(\mathcal{T})$ is described by 2 vectors $\mathbf{u}$ and $\mathbf{v}$ orthogonal to $\mathbf{w}$. The origin of $(\mathcal{T})$ is the projection of the point of coordinates $(0, 0, 0)$ on $(\mathcal{T})$ along $\mathbf{w}$. With spherical coordinate notations (see Fig. B.1), let

$$\mathbf{w} = [\sin \varphi \cos \theta, \sin \varphi \sin \theta, \cos \varphi]^t,$$

with $\varphi \in [0, \pi]$ and $\theta \in [0, 2\pi[$, and define $\mathbf{u}$ and $\mathbf{v}$ by

$$\mathbf{u} = [-\cos \varphi \cos \theta, -\cos \varphi \sin \theta, \sin \varphi]^t \text{ and } \mathbf{v} = [\sin \theta, -\cos \theta, 0]^t.$$

With this notation, the line $\mathcal{L}$ directed by $\mathbf{w}$ and which intersects the plane $(\mathcal{T})$ in $\mathbf{t} = u \mathbf{u} + v \mathbf{v}$ has the following equations:

$$\begin{aligned} u_{\theta, \varphi}(x, y, z) &= -x \cos \varphi \cos \theta - y \cos \varphi \sin \theta + z \sin \varphi = u, \\ v_{\theta, \varphi}(x, y, z) &= x \sin \theta - y \cos \theta = v, \end{aligned} \tag{B.16}$$

and the X-transform (B.1) relating $p(\mathbf{t}, \boldsymbol{\theta})$ and $f(x, y, z)$ can be written in function of $\mathbf{t}$, θ and φ by

$$\begin{aligned} p(\mathbf{t}, \theta, \varphi) &= \iiint_{\mathbb{R}^3} f(x, y, z) \delta_2 \left[\begin{pmatrix} u_{\theta, \varphi}(x, y, z) - u \\ v_{\theta, \varphi}(x, y, z) - v \end{pmatrix} \right] dx dy dz \\ &= \iiint_{\mathcal{P}} \delta_2 \left[(u_{\theta, \varphi}(x, y, z), v_{\theta, \varphi}(x, y, z))^t - \mathbf{t} \right] dx dy dz, \end{aligned} \quad (\text{B.17})$$

where δ_2 is the 2D Dirac distribution. Clearly, any square integrable function $F(\mathbf{t})$ satisfies the following relation:

$$\begin{aligned} \iint_{\mathbb{R}^2} p(\mathbf{t}, \theta, \varphi) F(\mathbf{t}) d\mathbf{t} &= \iiint_{\mathcal{P}} \left(\iint_{\mathbb{R}^2} F(\mathbf{t}) \delta_2 \left[\begin{pmatrix} u_{\theta, \varphi}(x, y, z) \\ v_{\theta, \varphi}(x, y, z) \end{pmatrix} - \mathbf{t} \right] d\mathbf{t} \right) dx dy dz \\ &= \iiint_{\mathcal{P}} F(u_{\theta, \varphi}(x, y, z), v_{\theta, \varphi}(x, y, z)) dx dy dz. \end{aligned}$$

Taking $F(\mathbf{t}) = F(u, v) = u^l v^q$ for any nonnegative integers l and q , we obtain an expression of the moments of the projections p . These moments are defined by

$$h_{lq}(\theta, \varphi) = \iint_{\mathbb{R}^2} u^l v^q p(\mathbf{t}, \theta, \varphi) d\mathbf{t}, \quad (\text{B.18})$$

and have the following expression:

$$h_{lq}(\theta, \varphi) = \iiint_{\mathcal{P}} (z \sin \varphi - x \cos \varphi \cos \theta - y \cos \varphi \sin \theta)^l (x \sin \theta - y \cos \theta)^q dx dy dz, \quad (\text{B.19})$$

that is, for $\varphi \neq \pi/2$ and $\theta \neq \pi/2 \bmod(\pi)$,

$$h_{lq}(\theta, \varphi) = \cos^l \varphi \cos^q \theta \iiint_{\mathcal{P}} (z \tan \varphi - x \cos \theta - y \sin \theta)^l (x \tan \theta - y)^q dx dy dz. \quad (\text{B.20})$$

One can obviously derive from (B.19) a relation between $h_{l, n-l}(\theta, \varphi)$ and the set $\{\mu_{ijk}, i + j + k = n\}$ of the n -th order geometric moments of $\mathcal{P}$:

$$\mu_{ijk} = \iiint_{\mathcal{P}} x^i y^j z^k dx dy dz, \quad i, j, k \geq 0. \quad (\text{B.21})$$

Note that the n -th order moments of the projections depend on only the n -th order geometric moments of the object.

The number of parameters μ_{ijk} such that $i + j + k = n$ is $\sum_{k=0}^n (n - k + 1) = (n + 1)(n + 2)/2$. Consequently, the linear system induced by (B.19) has $(n + 1)(n + 2)/2$ unknowns. To solve this system, one needs to have at least as much data $h_{lq}(\theta, \varphi)$, in the favorable case where the square matrix of the system is invertible. In other words, one needs at least $(n + 1)(n + 2)/2$ distinct values of $(l, n - l, \theta, \varphi)$ for the recovery of the variables μ_{ijk} , $i + j + k = n$. Since a fixed value of n yields $n + 1$ pairs (l, q) satisfying $n = l + q$, we deduce that $n + 1$ distinct values of l are available for any pair (θ, φ) . Hence, one should need at least $(n + 2)/2$ projections to determine all n -th order geometric moments of $\mathcal{P}$. As will be seen in the following sections, taking $(n + 2)/2$ projections does not lead to $(n + 1)(n + 2)/2$ independent equations. One actually has to select at least $n + 1$ projections to recover all n -th order geometric moments.

We hereafter give some further developments of relations (B.19) and (B.20), assuming the value of $n = l + q$ is fixed.

B.3.2 First step

For $l = 0$ (i.e., $q = n$), equation (B.19) yields for all φ in $[0, \pi[$ and θ in $[0, 2\pi[$:

$$h_{0n}(\theta, \varphi) = \iiint_{\mathcal{P}} (x \sin \theta - y \cos \theta)^n dx dy dz \quad (\text{B.22})$$

$$= \sum_{j=0}^n C_n^j (-1)^{n-j} \sin^j \theta \cos^{n-j} \theta \mu_{j,n-j,0} \quad (\text{B.23})$$

$$= \begin{cases} (-\cos \theta)^n \sum_{j=0}^n C_n^j (-\tan \theta)^j \mu_{j,n-j,0} & \text{if } \cos \theta \neq 0, \\ \mu_{n00} & \text{if } \theta = \pi/2, \\ (-1)^n \mu_{n00} & \text{if } \theta = 3\pi/2. \end{cases} \quad (\text{B.24})$$

Expression (B.23) is very similar to (B.9).

— If all angles θ_i are distinct *modulo* π , and none of them is equal to $\pi/2$ *modulo* π , (B.24) yields

$$\mathbf{h}_0^n = (-1)^n \mathcal{D}_1^n(\boldsymbol{\theta}) \begin{pmatrix} 1 & \tan(\theta_0) & \cdots & \tan^n(\theta_0) \\ 1 & \tan(\theta_1) & \cdots & \tan^n(\theta_1) \\ \vdots & \vdots & & \vdots \\ 1 & \tan(\theta_n) & \cdots & \tan^n(\theta_n) \end{pmatrix} \mathcal{C}_2^n \boldsymbol{\mu}_0^n, \quad (\text{B.25})$$

where the operator $\mathcal{D}_1^n$ has been defined in (B.13), and:

$$\mathcal{C}_2^n = \text{diag} [C_n^0, -C_n^1, \dots, (-1)^n C_n^n], \quad \mathbf{h}_0^n = \begin{pmatrix} h_{0n}(\theta_0, \varphi_0) \\ h_{0n}(\theta_1, \varphi_1) \\ \vdots \\ h_{0n}(\theta_n, \varphi_n) \end{pmatrix} \quad \text{and} \quad \boldsymbol{\mu}_0^n = \begin{pmatrix} \mu_{0n0} \\ \mu_{1,n-1,0} \\ \vdots \\ \mu_{n00} \end{pmatrix}. \quad (\text{B.26})$$

Similarly to Section B.2, system (B.25) is invertible.

— If all angles θ_i are distinct *modulo* π and one of them is equal to $\pi/2$ *modulo* π , the situation is also similar to Section B.2; it is possible to compute the moments $\mu_{j,n-j,0}$, $j = 0, \dots, n$ from the $n+1$ projections (θ_i, φ_i) , by estimating μ_{n00} in a first step.

— If several angles θ_i are equal *modulo* π , formula (B.24) produces the same equation for the corresponding indexes.

In conclusion, given m projections (θ_i, φ_i) ($m \geq n+1$), it is possible to recover $\boldsymbol{\mu}_0^n$ if and only if there exists $n+1$ distinct values (*modulo* π) of the angles θ_i , whatever the values of the angles φ_i .

B.3.3 Following steps

We proceed recursively on l and we use at step l a total number of projections which is equal or greater than $n-l+1 = q+1$. As before, we consider the $q+1$ first projections (θ_j, φ_j) , $j = 0, \dots, q$. We assume having computed $\boldsymbol{\mu}_0^n, \boldsymbol{\mu}_1^n, \dots, \boldsymbol{\mu}_{l-1}^n$ where

$$\boldsymbol{\mu}_i^n = [(\mu_{0,n-i,i}), (\mu_{1,n-i-1,i}), \dots, (\mu_{n-i,0,i})]^t, \quad i = 0, \dots, l-1. \quad (\text{B.27})$$

We wish to determine $\boldsymbol{\mu}_l^n$, whose size is $q+1$, from the data $h_{lq}(\theta_j, \varphi_j)$, $j = 0, \dots, q$. Developing the first factor of the expression in the integral (B.19) in function of the powers of $(z \sin \varphi)$ and $(x \cos \varphi \cos \theta + y \cos \varphi \sin \theta)$ gives, for all (θ, φ) :

$$h_{lq}(\theta, \varphi) = \sum_{i=0}^{l-1} h_{lq}^i(\theta, \varphi) + \sin^l \varphi \iiint_{\mathcal{P}} z^l (x \sin \theta - y \cos \theta)^q dx dy dz \quad (\text{B.28})$$

$$= \sum_{i=0}^{l-1} [\mathbf{a}_{li}^n(\theta, \varphi)]^t \boldsymbol{\mu}_i^n + \sin^l \varphi \iiint_{\mathcal{P}} z^l (x \sin \theta - y \cos \theta)^q dx dy dz, \quad (\text{B.29})$$

where $\mathbf{a}_{il}^n(\theta, \varphi)$ is a vector of size $n - i + 1$, whose elements correspond to z^i in the development of $h_{lq}(\theta, \varphi)$ (in fact, the p -th element of $\mathbf{a}_{il}^n(\theta, \varphi)$ corresponds to $\mu_{p, n-i-p, i}$), and $h_{lq}^i(\theta, \varphi) = [\mathbf{a}_{il}^n(\theta, \varphi)]^t \boldsymbol{\mu}_i^n$. It is obvious from (B.19) that $h_{lq}^i(\theta, \varphi)$ is of the form

$$h_{lq}^i(\theta, \varphi) = (-1)^{n-i} C_l^i \sin^i \varphi \cos^{l-i} \varphi g_i^{ln}(\theta, \varphi), \quad (\text{B.30})$$

where $g_i^{ln}(\theta, \varphi)$ is given by:

$$\begin{aligned} g_i^{ln}(\theta, \varphi) &= \iiint_{\mathcal{P}} \left\{ z^i (x \cos \theta + y \sin \theta)^{l-i} (-x \sin \theta + y \cos \theta)^q \right\} dx dy dz \\ &= \iiint_{\mathcal{P}} z^i \sum_{k=0}^q \left\{ (-1)^k C_q^k \sin^k \theta \cos^{q-k} \theta \times \right. \\ &\quad \left. x^k y^{q-k} (x \cos \theta + y \sin \theta)^{l-i} \right\} dx dy dz \\ &= \iiint_{\mathcal{P}} z^i \sum_{k=0}^q \sum_{j=0}^{l-i} \left\{ (-1)^k C_q^k C_{l-i}^j \sin^{k+l-i-j} \theta \cos^{q-k+j} \theta \times \right. \\ &\quad \left. x^{k+j} y^{q-k+l-i-j} \right\} dx dy dz \\ &= \sum_{k=0}^q \sum_{j=0}^{l-i} \left\{ (-1)^k C_q^k C_{l-i}^j \sin^{k+l-i-j} \theta \cos^{q-k+j} \theta \mu_{k+j, n-k-i-j, i} \right\}. \end{aligned} \quad (\text{B.31})$$

For all i in $\{0, \dots, l-1\}$, for all p in $\{0, \dots, n-i\}$, the p -th element of $\mathbf{a}_{il}^n(\theta, \varphi)$, which corresponds to $\mu_{p, n-i-p, i}$, can be deduced from expressions (B.30) and (B.31):

$$\begin{aligned} [\mathbf{a}_{il}^n(\theta, \varphi)]_p &= (-1)^{n-i} C_l^i \sin^i \varphi \cos^{l-i} \varphi \times \\ &\quad \sum_{k=\max(0, i+p-l)}^{\min(q, p)} (-1)^k C_q^k C_{l-i}^{p-k} \sin^{2k-p+l-i} \theta \cos^{p+q-2k} \theta. \end{aligned} \quad (\text{B.32})$$

We here define the transformed moments of the projections by

$$h_{lq}(\theta, \varphi) = \sum_{i=0}^{l-1} h_{lq}^i(\theta, \varphi), \quad (\text{B.33})$$

and the vector $\mathbf{h}_l^n$ of these transformed moments, of size $q+1$, whose j -th element is $h_{lq}(\theta_j, \varphi_j) - \sum_{i=0}^{l-1} h_{lq}^i(\theta_j, \varphi_j)$. Because of the similarity between equations (B.22) and (B.29), we proceed as in the case $l=0$ seen in Section B.3.2 to obtain the linear system relating $\mathbf{h}_l^n$ and $\boldsymbol{\mu}_l^n$. We obtain:

$$\mathbf{h}_l^n = (-1)^q \mathcal{D}_2^{lq}(\boldsymbol{\varphi}) \left((\sin^j(\theta_i) \cos^{q-j}(\theta_i))_{i,j} \right) \mathcal{C}_2^q \boldsymbol{\mu}_l^n, \quad (\text{B.34})$$

where $\mathcal{C}_2^q$ has been introduced in (B.26), and $\mathcal{D}_2^{lq}(\boldsymbol{\varphi}) = \text{diag}[\sin^l(\varphi_0), \sin^l(\varphi_1), \dots, \sin^l(\varphi_q)]$. Note that the linear system (B.34) is totally equivalent to those derived in Section B.2. Consequently, (B.34) is also invertible if all angles θ_i are distinct *modulo* π and none of the angles φ_i is equal to 0 *modulo* π . Indeed, one can clearly remark from equation (B.29) that a null value of $\sin(\varphi)$ yields no information on $\boldsymbol{\mu}_l^n$, unlike the case $l=0$.

In conclusion, it is possible to recover without any ambiguity the vector $\boldsymbol{\mu}_l^n$ from m projections (θ_i, φ_i) ($m \geq n+1$) and the computed moments $\boldsymbol{\mu}_0^n, \dots, \boldsymbol{\mu}_{l-1}^n$ if and only if there exists $n-l+1$ indexes $i_0, \dots, i_{n-l}$ such that all angles θ_{i_j} are distinct *modulo* π and none of the corresponding angles φ_{i_j} is equal to 0 *modulo* π .

B.3.4 Conclusion

The final conclusion of Section B.3 is that it is possible to recover the moments $\mu_0^n, \dots, \mu_n^n$ from m projections (θ_i, φ_i) ($m \geq n + 1$) if and only if there are $n + 1$ projections (say $(\theta_{i_j}, \varphi_{i_j}), j = 0, \dots, n$) such that:

- Not more than one of the angles φ_{i_j} is equal to 0 *modulo* π .
- All angles θ_{i_j} corresponding to non null values of φ_{i_j} (*modulo* π) are distinct *modulo* π .

B.4 Proposed Algorithms

We can derive from Sections B.2 and B.3 two algorithms which afford the estimation of the geometric moments of the object up to the n -th order from a finite number of projections.

In fact, in both 2D and 3D cases, we have obtained linear relations between the geometric moments of the object and those of the projections. Then, we have established the necessary conditions for their exact inversion. These results can be easily extended to the case in which the data are not the exact projections of the object (noisy data) or the case in which the number of projections is different from the necessary one (under or over-determined system). In case of noisy data and over-determined system, we define a solution in the least squares (LS) sense. And in the under-determined case, we define a solution in the minimum norm least squares (MNLS) sense. In the following, we denote m the number of available projections.

B.4.1 2D case

The 2D algorithm is the following, given the projections $\theta_0, \dots, \theta_{m-1}$:

For all n in $\{0, \dots, m-1\}$,

1. — Using (B.6), compute the moments of the projections $h_n(\theta_j)$, $j = 0, \dots, m-1$;
— Define the vector $\mathbf{h}^n = [h_n(\theta_0), h_n(\theta_1), \dots, h_n(\theta_{m-1})]^t$.
2. — If none of the angles θ_j is equal to $\pi/2$, then solve equation (B.12);
— Else, first deduce μ_{0n} then construct the vectors (B.14) and solve (B.15).

Remarks:

- Although the matrices in relations (B.12) and (B.15) are theoretically invertible, they are ill-conditioned if any two projection angles have very close values. In this case, we may define a regularized solution. However, this is not the main subject of this paper.
- Similar inversion problems occur in antenna array processing (AAP) (Basu and Bresler, 1997; Mohammad-Djafari and Khayati, 1997a,b). In this field, one has to inverse a system such (B.12), where $\mathcal{D}_1^n(\boldsymbol{\theta})$ and $\mathcal{C}_1^n$ are replaced by the identity matrices. The variables θ_i stand for the directions of arrival of the sources, $\mathbf{h}^n$ are the data observed by the antenna array and $\boldsymbol{\mu}^n$ are the relative source amplitudes. However, the main objective is here to recover $\theta_i, i = 0, \dots, n$ and then $\boldsymbol{\mu}^n$. In this application, it has been shown that, thanks to the particular structure of the matrix relating $\mathbf{h}^n$ and $\boldsymbol{\mu}^n$ (Van Der Monde matrix), it is possible to recover both the directions θ_i and the vector $\boldsymbol{\mu}^n$, assuming all directions θ_i are different. We believe that our 3D extension may interest the AAP research.

B.4.2 3D case

From Section B.3, we derive straightforwardly an algorithm to reconstruct the first order geometric moments of an object from its projections.

For $l = 0$, all the data are used to compute $\boldsymbol{\mu}_0^n$. For other values of l , the algorithm takes into account the data (θ_i, φ_i) such that φ_i is not equal to 0 *modulo* π (say p data, $p \geq n$), since the others provide no information on $\boldsymbol{\mu}_l^n$. At step l , $\mathbf{h}_l^n$ is a vector of size p , $\mathcal{D}_1^n(\boldsymbol{\theta})$ and $\mathcal{D}_2^{lq}(\boldsymbol{\varphi})$ are $p \times p$ matrices and the Van Der Monde matrices in relations (B.25) and (B.34) are $p \times (n+1)$ and $p \times (q+1)$. The algorithm is the following:

For all $n \geq 0$,

1. Check that the m data satisfy the condition stated in Section B.3.4.
2. Compute $h_{lq}(\theta_j, \varphi_j)$ for $j = 0, \dots, m-1$ and for all (l, q) such that $l+q = n$.
3. Do $l = 0$.
 - Define $\mathbf{h}_0^n = [h_{0n}(\theta_0, \varphi_0), h_{0n}(\theta_1, \varphi_1), \dots, h_{0n}(\theta_{m-1}, \varphi_{m-1})]^t$;
 - If none of the angles θ_j is equal to $\pi/2$ *modulo* π , then solve equation (B.25);
 - Else, proceed as in 2D case to compute μ_{n00} and then $[(\mu_{0n0}), \dots, (\mu_{n-1,1,0})]^t$.
4. For $l = 1, \dots, n$,
 - Compute $\mathbf{h}_l^n$ using relations (B.32) and (B.33). $\mathbf{h}_l^n$ is here a $m \times 1$ vector;
 - Discard from vector $\mathbf{h}_l^n$ the data $h_{lq}(\theta_j, \varphi_j)$ corresponding to null values of φ_j (*modulo* π). $\mathbf{h}_l^n$ is here a $p \times 1$ vector;
 - Inverse the linear relation (B.34) considering the case in which some of the angles θ_j are equal to $\pi/2$ *modulo* π , as performed in step 3.

B.5 Simulation and numerical results

To show the performances of the proposed method, we consider objects for which the exact values of the first order geometric moments can be computed either analytically or numerically. Since ellipsoids are defined from their geometric moments up to the second order, and any geometric moment of a polyhedral object can be computed from the coordinates of its vertices (Mirtich, 1996), we have chosen these two types of objects. Hence, in the following, we consider the tomographic projection data of an ellipsoid and the ones of a specific polyhedron along 4 directions. To take into account the measurement noise, we have added zero-mean, white Gaussian random variables to the projections. In the following, we reconstruct the geometric moments of the polyhedron up to the second order and the ones of the ellipsoid up to the third, since the theoretical computation of the third order moments of an ellipsoid is straightforward.

Both objects are plotted on Fig. B.2 and their noisy projections on Figs. B.3 and B.4. These plots correspond to signal to noise ratios (SNR) equal to 20, 0 and -10 dB. The SNR is defined by

$$\text{SNR} = 10 \log(v_p/v_b), \quad (\text{B.35})$$

where v_p is the empirical variance of the noiseless projections and v_b is that of the added noise. The values of (θ_j, φ_j) , $j = 0, \dots, 3$ corresponding to these projections are $\varphi_j = \pi/2$ and $\theta_j = j\pi/4$. This means that all the projection planes are parallel to the z axis. Note that on Figs. B.3 and B.4, the horizontal axis of the j -th projection corresponds to the vector $\mathbf{u}$ associated with (θ_j, φ_j) and the vertical axis corresponds to $\mathbf{v}$. For $\varphi = \pi/2$, we have $\mathbf{u} = \mathbf{z}$. Hence, the horizontal axes of each projections are identical to the $\mathbf{z}$ axis.

From these data, using the proposed algorithm, we have computed the geometric moments of the objects:

- In the polyhedral case, the direct numerical computation of the geometric moments from the

	μ_{000}	μ_{100}	μ_{010}	μ_{001}
Exact values	0.0099	0.0049	0.0046	0.0062
SNR =20	0.0100	0.0049	0.0046	0.0063
SNR =0	0.0101	0.0050	0.0047	0.0065
SNR =-10	0.0116	0.0051	0.0043	0.0069

	μ_{200}	μ_{020}	μ_{002}	μ_{011}	μ_{101}	μ_{110}
Exact values	0.0024	0.0022	0.0041	0.0029	0.0031	0.0023
SNR =20	0.0024	0.0022	0.0041	0.0029	0.0031	0.0023
SNR =0	0.0025	0.0023	0.0043	0.0030	0.0032	0.0024
SNR =-10	0.0028	0.0016	0.0044	0.0025	0.0029	0.0019

Table B.1: Polyhedron - Reconstruction of the moments up to the second order.

coordinates of the vertices of the polyhedron is done using analytical expressions and algorithms proposed by Mirtich (1996). The comparison of the results obtained for different values of the SNR and the real ones is done in Table B.1. It is interesting to notice that despite the fact that the data are very noisy, our algorithm provides an accurate reconstruction of the geometric moments.

— In the ellipsoidal case, the real values of all the first and third order moments are equal to 0 since it is centered at the origin. And the second order moments are easily obtained from the characteristic parameters of the ellipsoid. As in the previous case, the reconstructed results for the same values of the SNR are plotted in Tables B.2 and B.3.

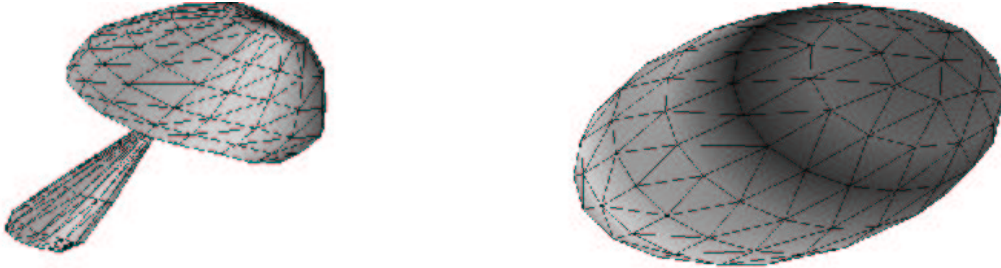


Figure B.2: The polyhedral object and the ellipsoid.

B.6 Conclusion

We considered the problem of the estimation of the geometric moments of a compact object from a few number of its X-ray projections. The proposed algorithm involves two steps: the computation of the moments of the projections from the data and then the estimation of the geometric moments of the object. This second step involves the inversion of a linear system for which we established the necessary conditions to obtain a unique solution in both 2D and 3D cases. The performances of the proposed algorithm are compared to the direct computation of the actual values of the moments for two classes of objects: ellipsoidal shapes and polyhedral shapes. The proposed algorithm seems to be very robust against the errors on the data.

	μ_{000}	μ_{100}	μ_{010}	μ_{001}
Exact values	1.2566	0.0000	0.0000	0.0000
SNR =20	1.2562	-0.0014	0.0009	0.0038
SNR =0	1.2934	-0.0229	-0.0241	-0.0186
SNR =-10	1.2889	-0.1773	0.0059	-0.0444

	μ_{200}	μ_{020}	μ_{002}	μ_{011}	μ_{101}	μ_{110}
Exact values	0.0974	0.1414	0.1634	0.0762	0.0440	0.0381
SNR =20	0.0982	0.1429	0.1643	0.0730	0.0425	0.0386
SNR =0	0.1344	0.1089	0.1777	0.0924	0.0496	0.0108
SNR =-10	0.0914	0.1482	0.1599	-0.0126	0.0131	0.0340

Table B.2: Ellipsoid - Reconstruction of the moments up to the second order.

	μ_{300}	μ_{210}	μ_{120}	μ_{030}
Exact values	0.0000	0.0000	0.0000	0.0000
SNR =20	-0.0006	-0.0002	0.0006	0.0037
SNR =0	0.0126	0.0048	-0.0255	-0.0415
SNR =-10	-0.1620	-0.0826	0.0177	0.1778

	μ_{201}	μ_{111}	μ_{021}	μ_{102}	μ_{012}	μ_{300}
Exact values	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
SNR =20	0.0043	0.0001	-0.0011	-0.0026	0.0034	0.0045
SNR =0	-0.0383	-0.0265	-0.0036	-0.0169	-0.0430	-0.0269
SNR =-10	-0.0100	-0.0247	-0.0655	-0.1333	0.0703	-0.0763

Table B.3: Ellipsoid - Reconstruction of the third order moments.

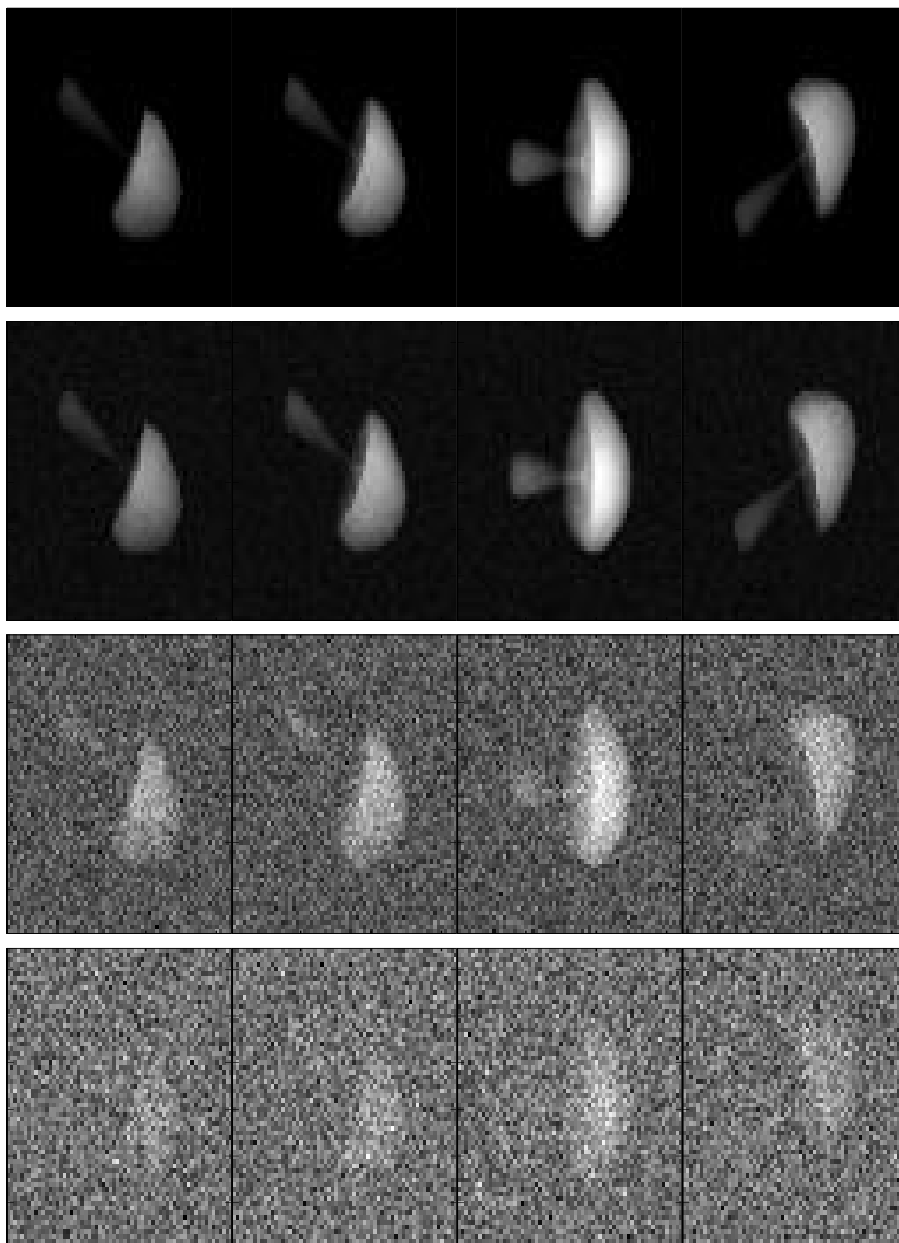


Figure B.3: Noiseless and noisy projections of the polyhedron; the three last rows correspond to SNR equal to 20, 0 and -10 dB. Each row contains four projections corresponding to four different directions (one projection per column).

References

- Y. Amit and K. M. Manbeck. Deformable template models for emission tomography. *IEEE Trans. Medical Imaging*, 12(2):260–268, June 1993.
- S. Basu and Y. Bresler. Tomography with unknown view angles. In *Proc. IEEE ICASSP*, volume 4, pages 2845–2848, Munich, Germany, April 1997.
- X. L. Battle, G. S. Cunningham, and K. M. Hanson. Tomographic reconstruction using 3D deformable models. *Phys. Med. Biol.*, 43:983–990, 1998.
- J. E. Besag. On the statistical analysis of dirty pictures (with discussion). *J. R. Statist. Soc. B*, 48(3):259–302, 1986.

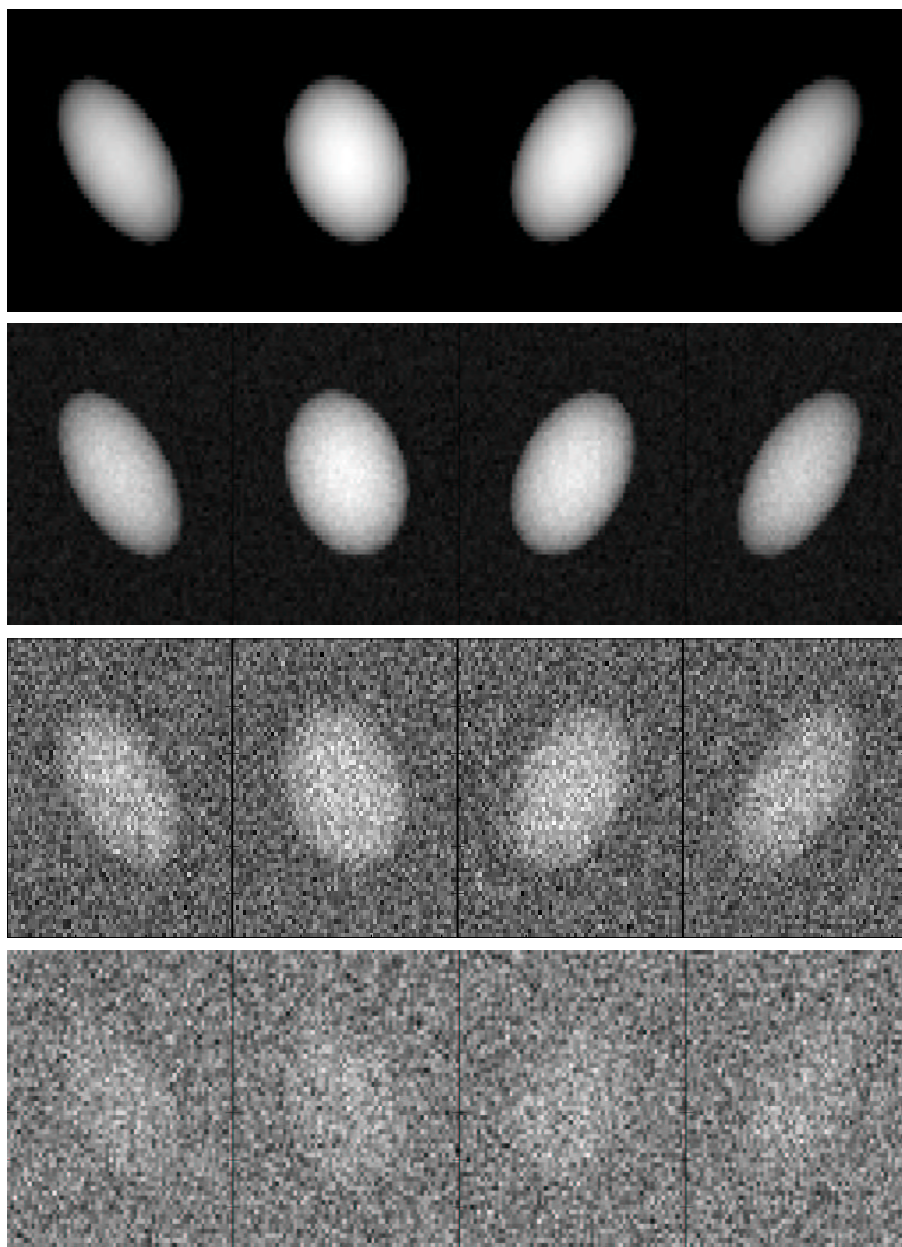


Figure B.4: Noiseless and noisy projections of the ellipsoid; the three last rows correspond to SNR equal to 20, 0 and -10 dB. Each row contains four projections.

J. E. Besag. Digital image processing: Towards Bayesian image analysis. *J. Appl. Statistics*, 16(3):395–407, 1989.

C. A. Bouman and K. D. Sauer. A generalized Gaussian image model for edge-preserving MAP estimation. *IEEE Trans. Image Processing*, 2(3):296–310, July 1993.

Y. Bresler and A. Macovski. Three-dimensional reconstruction from projections with incomplete and noisy data by object estimation. *IEEE Trans. Acoust. Speech, Signal Processing*, ASSP-35(8):1139–1152, August 1987.

K. M. Hanson, G. S. Cunningham, and R. J. McKee. Uncertainty assessment for reconstructions based on deformable geometry. *Int. J. Imaging Systems and Technology*, 8:506–512, 1997.

R. Malladi, J. A. Sethian, and B. C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(2):158–175, February 1995.

- P. Milanfar, W. C. Karl, and A. S. Willsky. A moment-based variational approach to tomographic reconstruction. *IEEE Trans. Image Processing*, 5(3):459–470, February 1996.
- P. Milanfar, G. C. Verghese, W. C. Karl, and A. S. Willsky. Reconstructing polygons from moments with connections to array processing. *IEEE Trans. Signal Processing*, 43(2):432–443, February 1995.
- B. Mirtich. Fast and accurate computation of polyhedral mass properties. *Journal of Graphic Tools*, 1(2):31–50, 1996.
- A. Mohammad-Djafari and Y. Khayi. Array processing techniques and shape reconstruction in tomography. In *Proc. IEEE ICASSP*, volume II, pages 3785–3789, Munich, Germany, April 1997a.
- A. Mohammad-Djafari and Y. Khayi. Évaluation de méthodes de traitement d’antenne en reconstruction d’image appliquée au contrôle non destructif. In *Actes du 16^e colloque GRETSI*, pages 135–139, Grenoble, France, September 1997b.
- M. Nikolova and A. Mohammad-Djafari. Eddy current tomography using a binary Markov model. *Signal Processing*, 49:119–132, 1996.
- D. J. Rossi and A. S. Willsky. Reconstruction from projections based on detection and estimation of objects – Parts I and II: Performance analysis and robustness analysis. *IEEE Trans. Acoust. Speech, Signal Processing*, ASSP-32(4):886–906, August 1984.
- F. Santosa. A level-set approach for inverse problems involving obstacles. *Control, Optimisation and Calculus of Variations*, 1:17–33, January 1996.
- K. D. Sauer and C. A. Bouman. Bayesian estimation of transmission tomograms using segmentation based optimization. *IEEE Trans. Nuclear Sciences*, 39:1144–1152, February 1992.
- W. Snyder, Y.-S. Han, G. Bilbro, R. T. Whitaker, and S. Pizer. Image relaxation: Restoration and feature extraction. *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-17(6):620–624, June 1995.
- M. Soumekh. Reconstruction from incomplete data in echo diffraction imaging. In *Proc. IEEE ICASSP*, pages 1304–1307, New York, NY, April 1988.

ANNEXE C

MÉTHODE POLYGONALE : DIFFÉRENCIATION DE $\mathcal{J}(\mathbf{v})$

Le critère régularisé à minimiser est défini sous forme compacte par $\mathcal{J}(\mathbf{v}) = \mathcal{K}(\mathbf{v}) + \lambda\mathcal{R}(\mathbf{v})$, où $\mathcal{R}(\mathbf{v})$ est un terme de douceur locale. Nous nous intéressons successivement à la différenciation de $\mathcal{K}$ et $\mathcal{R}$.

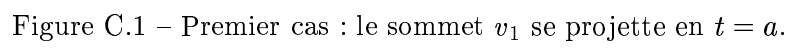
C.1 Terme d'adéquation aux données

Nous avons vu au § IV.4.4 que le calcul de $\nabla\mathcal{K}$ repose sur l'évaluation, si elles existent, des quantités $\partial p_{\mathbf{v}}(t)/\partial v_j$, pour une direction de projection θ fixée. Nous considérons le cas du premier sommet v_1 . Compte-tenu de la discussion sur la remise à jour des projections faite au § IV.4.3, nous avons le résultat suivant dans le cas non dégénéré¹ : $\partial p_{\mathbf{v}}(t)/\partial v_1$ existe *si et seulement si* $\partial p_{\mathbf{t}_1}(t)/\partial v_1$ est définie, et dans l'affirmative, ces quantités sont égales au signe près. Dans tout ce qui suit, nous considérons un triangle $\mathbf{t} = \{v_1, v_2, v_3\}$ non aplati, et nous exprimons sa projection orthogonale $p_{\mathbf{t}}(t)$ suivant $\mathbf{w}_{\theta-\pi/2}$. Comme illustré en figure C.1, la fonction $p_{\mathbf{t}}(t)$ est positive, continue (sauf cas particuliers) et affine par morceaux. Nous notons (c, b) son support et a la position de son maximum ($c \leq a \leq b$). $p_{\mathbf{t}}$ est donc affine sur les intervalles (c, a) et (a, b) et nulle ailleurs. Nous considérons les trois cas suivants, illustrés sur les figures C.1 et C.4 :

- (1) v_1 se projette en a , *i.e.*, $l = p_{\mathbf{t}}(v_1) \neq 0$,
- (2-a) v_1 se projette en b ,
- (2-b) v_1 se projette en c ,

et dans chaque cas, nous étudions les déplacements horizontal et vertical de $\vec{v}_1$. Dans la suite, nous notons a_0, b_0 , ou c_0 la position a, b , ou c de t qui correspond à la projection de v_1 , puis par a_1, b_1 , ou c_1 la position de t correspondant à la projection de $v_1 + dx_1$ ou $v_1 + i dy_1$ (voir la figure C.1).

¹Nous rappelons que dans le cas dégénéré où $v_1 \in (v_n, v_2)$, nous nous ramenons au cas de la projection de deux triangles ayant v_1 pour sommet en introduisant un sommet fictif.



C.1.1 Cas où $p_t(v_1) \neq 0$

Dans ce paragraphe, nous notons $\tilde{\mathbf{t}}$ la modification du triangle $\mathbf{t}$ suite à un déplacement de $\vec{v}_1$. Nous considérons chacun des intervalles $(c, a[$ et $]a, b)$ pour le calcul de la dérivée partielle complexe $\partial p_v(t)/\partial z_1 = \partial p_v(t)/\partial x_1 + i \partial p_v(t)/\partial y_1$.

Cas où v_1 se trouve en dessous de la droite (v_2, v_3) et d'un déplacement suivant (x)

— Si $c \leq t < a$ et si $\partial p_v(t)/\partial x_1$ existe, alors

$$\begin{aligned} \partial p_v(t)/\partial x_1 &= \lim_{dx_1 \rightarrow 0} (p_{\tilde{\mathbf{t}}}(t) - p_{\mathbf{t}}(t))/dx_1 \\ &= \lim_{dx_1 \rightarrow 0} \frac{t-c}{dx_1} \left(\frac{l_1}{a_1-c} - \frac{l_0}{a_0-c} \right) \\ &= \lim_{dx_1 \rightarrow 0} \frac{t-c}{dx_1(a_0-c)^2} (l_1(a_0-c) - l_0(a_1-c)). \end{aligned} \quad (\text{C.1})$$

Les deux positions a_0 et a_1 sont définies par $a_0 = \langle \vec{v}_1, \mathbf{w}_\theta \rangle$ et $a_1 = \langle \vec{v}_1 + dx_1 \mathbf{i}, \mathbf{w}_\theta \rangle = a_0 + dx_1 \cos \theta$. De plus, avec les notations de la figure C.2 (ϕ représente l'angle entre l'axe $\mathbf{i}$ et le segment (v_3, v_2)), nous avons $l_1 = l_0 + \delta_1 + \delta_2$ avec $\delta_1 = \tan(\phi - \theta) \cos \theta dx_1$ et $\delta_2 = \sin \theta dx_1$, soit finalement :

$$l_1 = l_0 + (\sin \theta - \cos \theta \tan(\theta - \phi)) dx_1 = l_0 + \sin \phi / \cos(\theta - \phi) dx_1. \quad (\text{C.2})$$

L'expression (C.1) devient alors :

$$\begin{aligned} \partial p_v(t)/\partial x_1 &= \lim_{dx_1 \rightarrow 0} \frac{t-c}{dx_1(a-c)^2} (l_1(a-c) - l_0(a-c + dx_1 \cos \theta)) \\ &= \frac{t-c}{a-c} (\sin \phi / \cos(\theta - \phi) - l \cos \theta / (a-c)) \\ &= [\sin \phi - d_{1,23} \cos \theta / (a-c)] (t-c) / [(a-c) \cos(\theta - \phi)], \end{aligned} \quad (\text{C.3})$$

en notant $d_{1,23} = l \cos(\theta - \phi)$ la distance de $\vec{v}_1$ à la droite (v_2, v_3) . Le cas $\theta = \phi [\pi/2]$ n'est pas envisagé car mène à $a = b = c$, compte-tenu de l'hypothèse (1).

— Si $a < r \leq b$, un calcul analogue mène à :

$$\partial p_v(t)/\partial x_1 = [\sin \phi + d_{1,23} \cos \theta / (b-a)] (b-t) / [(b-a) \cos(\theta - \phi)]. \quad (\text{C.4})$$

— En les autres valeurs de t , a exclu, la dérivée partielle est nulle.

La fonction $\partial p_v(t)/\partial x_1$ ne s'étend pas par continuité en $t = a$, sauf si $\theta = \pm \pi/2$, et dans ce cas, sa valeur est égale à 1. Les autres cas qui interviennent sous l'hypothèse (1) sont traités ci-après et illustrés en figure C.3.

Déplacement suivant la direction (y)

Nous remarquons que calculer la projection associée à un déplacement dy_1 dans la direction $\mathbf{j}$ équivaut à :

- Faire subir au repère $(\mathbf{i}, \mathbf{j})$ une rotation d'angle $\pi/2$;
- Effectuer un déplacement dy_1 dans la direction $\mathbf{i}$;
- Calculer la projection associée à ce déplacement sur le détecteur dirigé par $\mathbf{w}_{\theta-\pi/2}$.

L'expression de $\partial p_v(t)/\partial y_1$ s'obtient donc en retranchant $\pi/2$ à θ et ϕ dans (C.3) et (C.4) :

$$\frac{\partial p_v(t)}{\partial y_1} = \begin{cases} -(\cos \phi + d_{1,23} \sin \theta / (a-c)) (t-c) / [(a-c) \cos(\theta - \phi)] & \text{si } t \in (c, a[, \\ -(\cos \phi - d_{1,23} \sin \theta / (b-a)) (b-t) / [(b-a) \cos(\theta - \phi)] & \text{si } t \in]a, b), \\ 0 & \text{si } r < c \text{ ou } r > b. \end{cases} \quad (\text{C.5})$$

Cette dérivée n'est définie en a que si $\theta = 0$, et vaut alors -1.

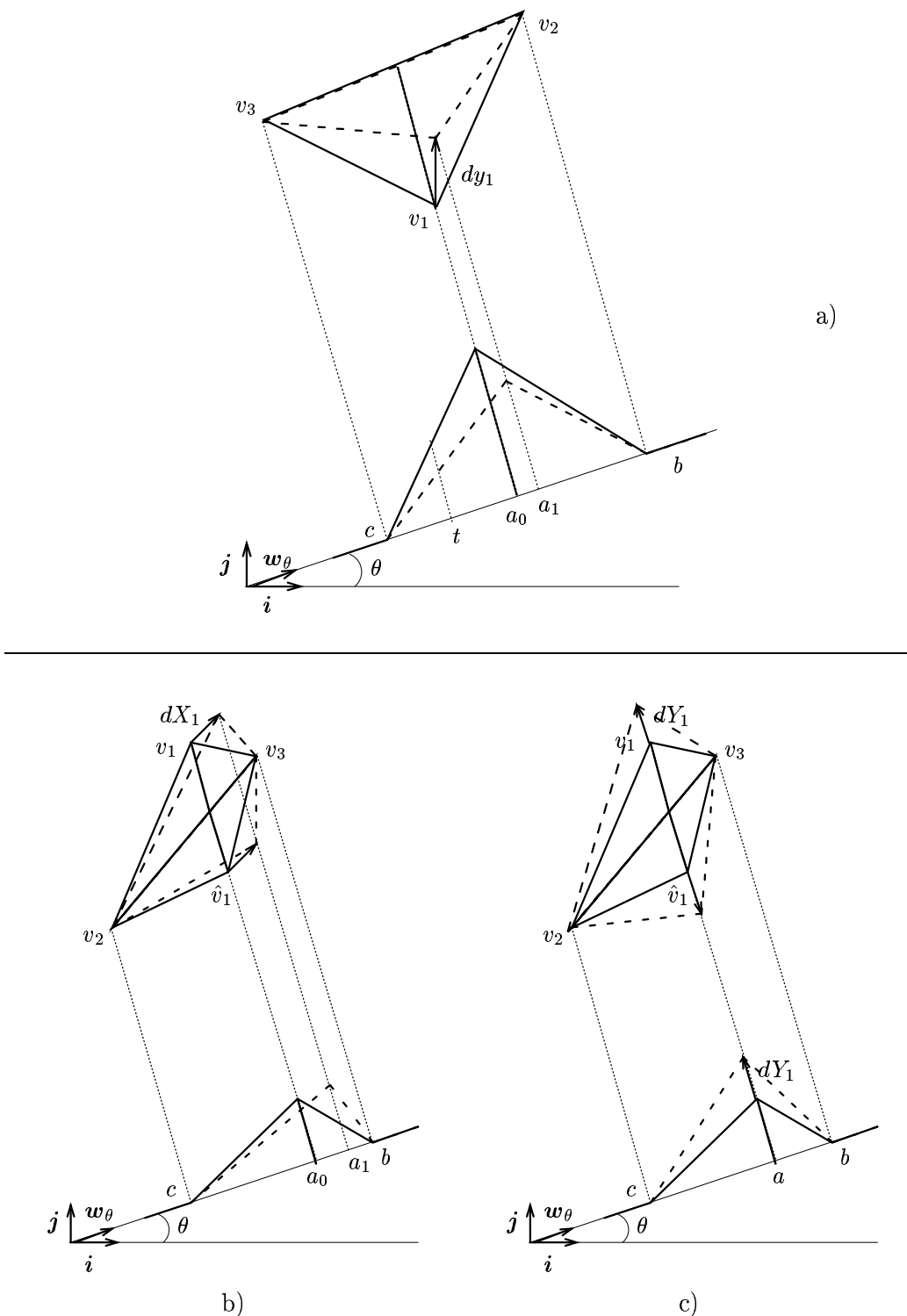


Figure C.3 – Situations dérivant du premier cas : a) Modification de $\vec{v}_1$ dans la direction (y) , b) et c) Sommet v_1 situé au dessus de la droite (v_2v_3) . Cas de déplacements suivant $\mathbf{w}_\phi (l_0 = l_1)$ et $\mathbf{w}_{\theta+\pi/2} (a_0 = a_1)$.

Cas où v_1 se trouve au dessus de la droite (v_2, v_3)

Dans ce cas, illustré en figure C.3 (b,c), nous avons $\hat{v}_1 = v_1 - 2ip_t(a) \exp(i\theta)$ le symétrique de v_1 par rapport à (v_2, v_3) . En notant $\hat{t}$ le triangle $\{\hat{v}_1, v_3, v_2\}$, nous déduisons les expressions analytiques des dérivées partielles de $p_t(t)$ de celles de $p_{\hat{t}}(t)$, en remarquant que ces deux triangles ont la même projection orthogonale sur $\mathbf{w}_\theta$, et que :

- un déplacement de $\vec{v}_1$ suivant $\mathbf{w}_\phi$ « équivaut » au même déplacement de $\vec{\hat{v}}_1$ (voir la figure C.3 (b)) ;
- un déplacement de $\vec{v}_1$ suivant $\mathbf{w}_{\theta+\pi/2}$ « équivaut » au déplacement opposé de $\vec{\hat{v}}_1$ suivant la même direction (voir la figure C.3 (c)).

Nous définissons donc le changement de variables qui transforme $\mathbf{i}$ et $\mathbf{j}$ en $\mathbf{w}_\phi$ et $\mathbf{w}_{\theta+\pi/2}$:

$$(x_1, y_1)^t = \mathbf{R}(\theta, \phi) (X_1, Y_1)^t, \text{ où } \mathbf{R}(\theta, \phi) = \begin{pmatrix} \cos \phi & -\sin \theta \\ \sin \phi & \cos \theta \end{pmatrix}. \quad (\text{C.6})$$

Ce changement de variables est proprement défini si $\det \mathbf{R}(\theta, \phi) = \cos(\theta - \phi) \neq 0$. Or, nous avons vu précédemment que le cas $\theta = \phi \in [\pi/2]$ n'est pas compatible avec l'hypothèse (1). Le changement de variables (C.6) est donc licite et nous avons pour $t \neq a$:

$$\begin{aligned} (\partial p_t(t)/\partial x_1, \partial p_t(t)/\partial y_1)^t &= \mathbf{R}(\theta, \phi)^{-t} (\partial p_t(t)/\partial X_1, \partial p_t(t)/\partial Y_1)^t \\ &= \mathbf{R}(\theta, \phi)^{-t} (\partial p_{\hat{t}}(t)/\partial \hat{X}_1, -\partial p_{\hat{t}}(t)/\partial \hat{Y}_1)^t \\ &= \mathbf{S}(\theta, \phi) (\partial p_{\hat{t}}(t)/\partial \hat{x}_1, \partial p_{\hat{t}}(t)/\partial \hat{y}_1)^t, \end{aligned} \quad (\text{C.7})$$

où $\hat{x}_1, \hat{y}_1$ et $\hat{X}_1, \hat{Y}_1$ sont les coordonnées de $\hat{v}_1$ dans chaque repère, et

$$\begin{aligned} \mathbf{S}(\theta, \phi) &= \mathbf{R}(\theta, \phi)^{-t} \text{diag}[1, -1] \mathbf{R}(\theta, \phi)^t \\ &= \frac{1}{\cos(\theta - \phi)} \begin{pmatrix} \cos(\theta + \phi) & 2 \sin \phi \cos \theta \\ 2 \sin \theta \cos \phi & -\cos(\theta + \phi) \end{pmatrix}. \end{aligned} \quad (\text{C.8})$$

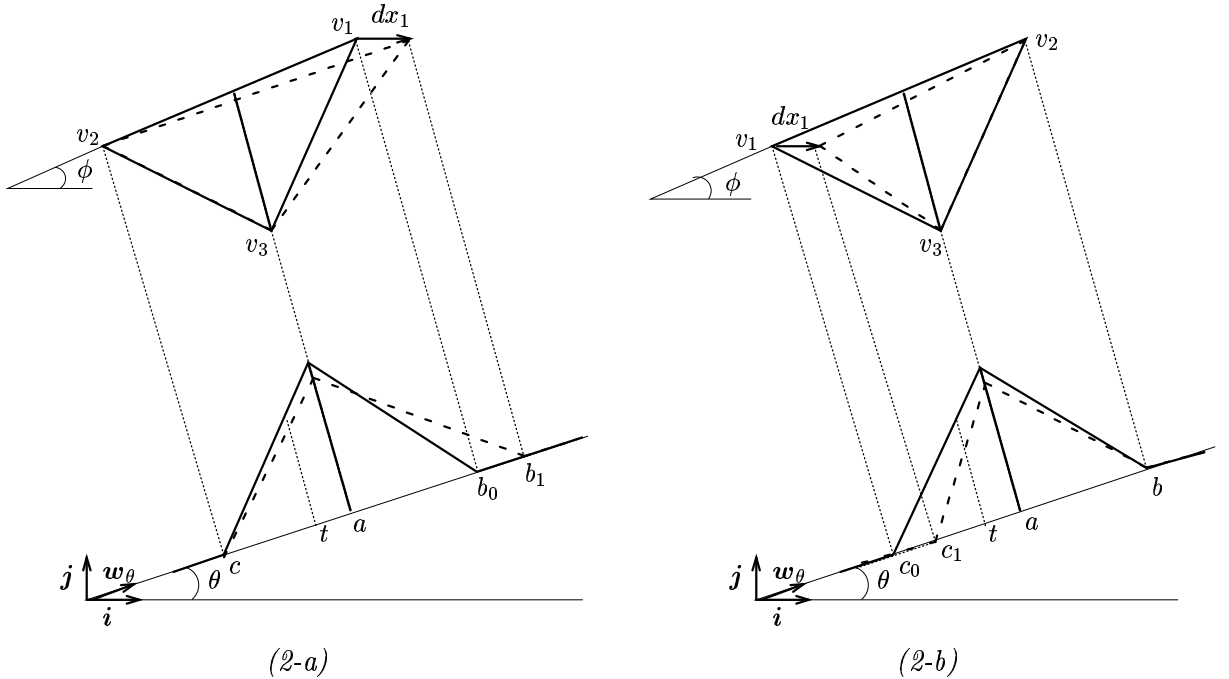
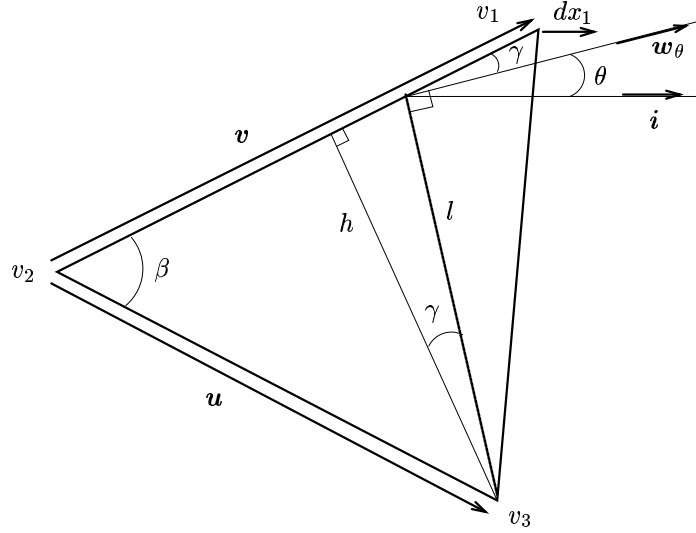


Figure C.4 – Cas où $p_t(v_1) = 0$. Projection de v_1 en $t = b$ (cas (2-a)) et $t = c$ (cas (2-b)).

Figure C.5 – Expression de $l = p_t(a)$ dans le cas où v_1 se projette en b .

C.1.2 Cas où $p_t(v_1) = 0$

— La fonction $p_t(t)$ a toujours pour expression :

$$p_t(t) = \begin{cases} l(t-c)/(a-c) & \text{si } t \in (c, a), \\ l(b-t)/(b-a) & \text{si } t \in (a, b), \\ 0 & \text{sinon.} \end{cases} \quad (\text{C.9})$$

En supposant que v_1 se projette en b et v_3 se projette en a^2 (voir les figures C.4 (a) et C.5), nous exprimons la longueur $l = p_t(t = a)$ en fonction des paramètres θ , $\mathbf{u} = \vec{v}_3 - \vec{v}_2$, $\mathbf{v} = \vec{v}_1 - \vec{v}_2$. En adoptant les notations de la figure C.5, nous avons $\cos \gamma = h/l$ et $h = \|\mathbf{u}\| |\sin \beta|$, où l'angle γ est défini par $\gamma = (\mathbf{w}_\theta, \mathbf{v})$. Nous en déduisons :

$$l = \|\mathbf{u}\| |\sin \beta| / \cos(\mathbf{w}_\theta, \mathbf{v}) = \frac{\varepsilon \det[\mathbf{u}, \mathbf{v}]}{\|\mathbf{v}\| \cos(\mathbf{w}_\theta, \mathbf{v})} = \frac{\varepsilon \det[\mathbf{u}, \mathbf{v}]}{\mathbf{v}^t \mathbf{w}_\theta},$$

où $\varepsilon = \pm 1 = \text{sgn}(\det[\mathbf{u}, \mathbf{v}])$. Le cas $\mathbf{v} \perp \mathbf{w}_\theta$ est exclu, car combiné à l'hypothèse (2-a), mène à un triangle $\mathbf{t}$ aplati ($c = a = b$). Facilement, nous avons $\partial \mathbf{v} / \partial v_1 = \mathbf{i} + i \mathbf{j}$, et nous déduisons :

$$\partial l / \partial v_1 = \varepsilon [(\mathbf{v}^t \mathbf{w}_\theta) iz(\mathbf{u}) - \det[\mathbf{u}, \mathbf{v}] \exp(i\theta)] / (\mathbf{v}^t \mathbf{w}_\theta)^2.$$

Nous évaluons de même qu'au § C.1.1, la quantité $\partial b / \partial v_1 = \exp(i\theta)$. Nous pouvons donc déduire de (C.9) l'expression analytique de la dérivée partielle de $p_t(t)$:

$$\frac{\partial p_t(t)}{\partial v_1} = \begin{cases} \partial l / \partial v_1 (t-c)/(a-c) & \text{si } t \in (c, a[, \\ \partial l / \partial v_1 (b-t)/(b-a) + l \exp(i\theta) (t-a)/(b-a)^2 & \text{si } t \in]a, b[, \\ 0 & \text{si } r < c \text{ ou } r > b. \end{cases} \quad (\text{C.10})$$

— Par continuité, $p_t(t)$ est dérivable par rapport à v_1 en $t = a$, et sa dérivée y prend la valeur $\partial l / \partial v_1$. $p_t(t)$ n'est dérivable par rapport à x_1 (resp. y_1) en $t = b$ que si $\theta = \pm \pi/2$ (resp. $\theta = 0$). Si $\theta = \pm \pi/2$, $\partial p_t(b) / \partial x_1 = 0$; si $\theta = 0$, $\partial p_t(b) / \partial y_1 = 0$.

²Le cas où v_1 se projette en b et v_2 se projette en a s'obtient en permutant v_2 et v_3 dans les expressions de ce paragraphe.

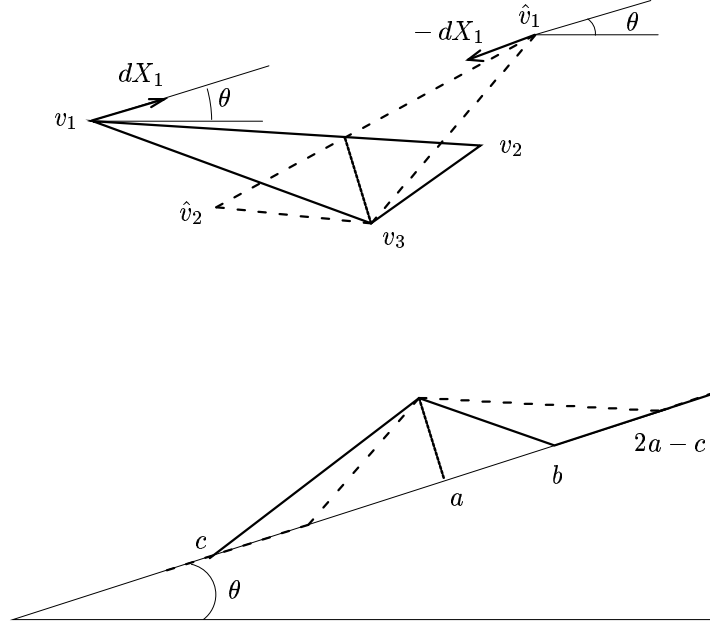


Figure C.6 – Passage du cas (2-a) au cas (2-b) par symétrie par rapport à la droite passant par v_3 et dirigée par $\mathbf{w}_{\theta+\pi/2}$.

— Le cas où v_1 se projette en c , illustré sur les figures C.4 (b) et C.5, se déduit du cas précédent. Nous supposons, sans perte de généralité, que v_3 est le sommet du triangle qui ne se projette pas en 0. En considérant le triangle $\hat{\mathbf{t}} = \{\hat{v}_1, \hat{v}_2, v_3\}$, symétrique de $\mathbf{t}$ par rapport à la droite passant par v_3 et dirigée par $\mathbf{w}_{\theta+\pi/2}$ (voir la figure C.6), nous avons $p_{\mathbf{t}}(t) = p_{\hat{\mathbf{t}}}(2a - t)$. Nous définissons le changement de variables correspondant à une rotation d'angle θ à partir de (C.6) :

$$(x_1, y_1)^t = \mathbf{R}(\theta, \theta) (X_1, Y_1)^t. \quad (\text{C.11})$$

Comme $\hat{X}_1 = 2a - X_1$ et $\hat{Y}_1 = Y_1$, $\partial p_{\mathbf{t}}(t)/\partial X_1 = -\partial p_{\hat{\mathbf{t}}}(2a - t)/\partial \hat{X}_1$ et $\partial p_{\mathbf{t}}(t)/\partial Y_1 = \partial p_{\hat{\mathbf{t}}}(2a - t)/\partial \hat{Y}_1$. Nous pouvons ainsi déduire, pour tout $t \neq c$:

$$\begin{aligned} (\partial p_{\mathbf{t}}(t)/\partial x_1, \partial p_{\mathbf{t}}(t)/\partial y_1)^t &= \mathbf{R}(\theta, \theta) (-\partial p_{\hat{\mathbf{t}}}(2a - t)/\partial \hat{X}_1, \partial p_{\hat{\mathbf{t}}}(2a - t)/\partial \hat{Y}_1)^t \\ &= -\mathbf{S}(\theta, \theta) (\partial p_{\hat{\mathbf{t}}}(2a - t)/\partial \hat{x}_1, \partial p_{\hat{\mathbf{t}}}(2a - t)/\partial \hat{y}_1)^t. \end{aligned} \quad (\text{C.12})$$

C.1.3 Cas particulier : $a = b$ ou $a = c$

Ce cas, dans lequel l'un des cotés contenant v_1 est orthogonal à $\mathbf{w}_{\theta}$, se ramène au cas (1). La projection est continue sur $\mathbb{R} - \{a\}$, affine sur (b, c) et nulle partout ailleurs. Sa dérivée partielle par rapport à v_1 est définie partout sauf en $t = a$, où $p_{\mathbf{t}}$ n'est pas définie. Son calcul est identique à celui effectué au § C.1.1, restreint à l'un des deux intervalles (c, a) ou (a, b) .

C.2 Terme de régularisation

En reprenant les notations du § IV.2.2.2, $\mathcal{R}(\mathbf{v})$ dépend des angles A_i . Il est défini par $\mathcal{R}(\mathbf{v}) = \sum_{i=1}^n \phi(\kappa_i(\mathbf{v}))$, où $\kappa_i(\mathbf{v}) = 1 + \cos A_i(\mathbf{v})$. Compte-tenu du caractère cyclique de la fonctionnelle $\mathcal{R}$, et du fait qu'à j fixé, v_j n'intervient que dans trois termes dont deux de nature similaire, il suffit d'évaluer les dérivées partielles de $\kappa_1(\mathbf{v})$ et $\kappa_2(\mathbf{v})$ par rapport à v_1 pour obtenir l'expression analytique du gradient de $\mathcal{R}$. Comme

$$\cos A_1(\mathbf{v}) = \cos A_1(\mathbf{t}_1) = (a_0^2 + a_2^2 - a_1^2)/(2a_0a_2),$$

en notant a_i les longueurs des arêtes opposées au sommet v_i dans le triangle $\mathbf{t}_1$ et $\partial a_i/\partial v_1 = (v_1 - v_{\bar{i}})/a_i$, où $\bar{i} = 2 - i$, nous avons $\partial(a_0a_2)/\partial v_1 = (v_1 - v_2)a_2/a_0 + (v_1 - v_0)a_0/a_2$, et finalement :

$$\partial \kappa_1(\mathbf{v})/\partial v_1 = 2(v_1 - v_1^*)/(a_0a_2) - \left[(v_1 - v_0)a_0^2 + (v_1 - v_2)a_2^2 \right] \cos A_1 / (a_0^2a_2^2). \quad (\text{C.13})$$

Pour $i = 0, 2$, a_i est indépendant de v_i , et nous avons :

$$\partial a_1^2/\partial v_i = 2(v_i - v_{\bar{k}}), \quad \partial a_{\bar{k}}^2/\partial v_i = 2(v_i - v_1) \quad \text{et} \quad \partial a_{\bar{k}}/\partial v_i = (v_i - v_1)/a_{\bar{k}},$$

donc finalement,

$$\partial \kappa_1(\mathbf{v})/\partial v_i = (v_{\bar{k}} - v_1)/(a_0a_2) - (v_i - v_1) \cos A_1 / a_{\bar{k}}^2. \quad (\text{C.14})$$

Ces résultats s'étendent en : $\partial \phi(\kappa_1(\mathbf{v}))/\partial v_i = \phi'(\kappa_1(\mathbf{v})) \partial \kappa_1(\mathbf{v})/\partial v_i$.

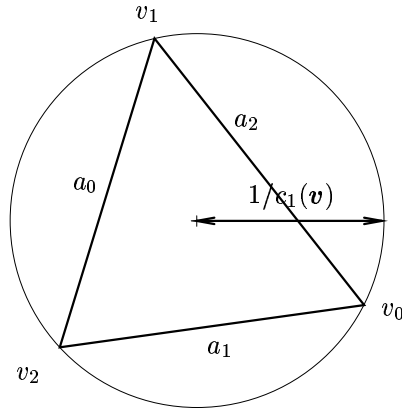


Figure C.7 – Courbure du triangle $\mathbf{t}_1$ — Notations.

ANNEXE D

MÉTHODE POLYÉDRIQUE : DIFFÉRENCIATION DE $\mathcal{K}(\mathbf{V})$

Nous avons vu au chapitre V qu'en définissant un polyèdre à n sommets $\mathbf{V} = \{V_1, \dots, V_n\}$, $V_i \in \mathbb{R}^3$ et à facettes fixées, nous pouvons définir un critère pénalisé sous la forme :

$$\mathcal{J}(\mathbf{V}) = \mathcal{K}(\mathbf{V}) + \lambda \mathcal{R}(\mathbf{V}). \quad (\text{D.1})$$

Le terme d'adéquation aux données s'exprime par :

$$\mathcal{K}(\mathbf{V}) = \|\mathbf{p} - \mathcal{A}(\mathbf{V})\|^2, \quad (\text{D.2})$$

où $\mathcal{A}$ représente l'opérateur de projection. Le terme de régularisation est défini par la somme de quantités locales à chaque sommet. Nous étudions ici l'évaluation de $\nabla \mathcal{K}$, celle de $\nabla \mathcal{R}$, plus simple, étant traitée dans l'annexe E.

D.1 Introduction

Nous avons vu au chapitre V que le calcul exact de $\nabla \mathcal{K}(\mathbf{V})$ se déduit du calcul exact de la dérivée partielle par rapport à un sommet, par exemple V_1 , d'une projection quelconque d'un tétraèdre $\mathcal{T}$ dont V_1 est l'un des sommets. Dans ce qui suit, nous considérons un tétraèdre $\mathcal{T} = \{V_i, i = 1, \dots, 4\}$ et sa projection parallèle orthogonale $\mathcal{A}_{uv}(\mathcal{T})$ de direction $\mathbf{w}$ et d'angle sphérique $\boldsymbol{\theta}$. Comme indiqué en figure I.4 (page 18), nous repérons le plan détecteur par les deux directions $\mathbf{u}$ et $\mathbf{v}$ orthogonales à $\mathbf{w}_{\boldsymbol{\theta}}$, et définies par l'équation (I.13). Ce plan est alors décrit par deux paramètres (u, v) , chaque mesure en $\mathbf{t} = u \mathbf{u} + v \mathbf{v}$ correspond à l'intégrale sur la droite d'équations $\mathbf{t}_{\boldsymbol{\theta}}(x, y, z) = \mathbf{t}$, où la fonction $\mathbf{t}_{\boldsymbol{\theta}}$ est définie en équation (I.15).

Nous réalisons dans un premier temps un changement de repère pour nous ramener au cas de la projection $p(x, y) = \mathcal{A}_{xy}(\mathcal{T})$ suivant la direction $\mathbf{k}$ de la verticale, puis nous étudions la projection de $\mathcal{T}$ proprement dite. Puis, nous déduisons au § D.4 les conditions de la différentiabilité de $\mathcal{K}$, en exprimant l'influence de chaque tétraèdre $\mathcal{T}$ sur de $\nabla \mathcal{K}$. Enfin, au § D.6, nous généralisons ces résultats au cas où les projections sont parallèles mais non orthogonales au détecteur.

D.2 Changement de repère

Pour ramener le problème à la projection suivant la direction verticale $\mathbf{k}$ sur un plan $z = \text{constante}$, nous effectuons une rotation qui transforme les vecteurs $\mathbf{u}$, $\mathbf{v}$ et $\mathbf{w}$ en $\mathbf{i}$, $\mathbf{j}$ et $\mathbf{k}$ respectivement. Les formules de changement de coordonnées sont :

$$(u, v, w)^t = \mathbf{M}^{-1}(\boldsymbol{\theta}) (x, y, z)^t, \quad \text{où } \mathbf{M}(\boldsymbol{\theta}) = \begin{pmatrix} -\cos \theta \cos \varphi & \sin \theta & \cos \theta \sin \varphi \\ -\sin \theta \cos \varphi & -\cos \theta & \sin \theta \sin \varphi \\ \sin \varphi & 0 & \cos \varphi \end{pmatrix}, \quad (\text{D.3})$$

et (u, v, w) et (x, y, z) décrivent le même point dans les repères $(\mathbf{u}, \mathbf{v}, \mathbf{w})$ et $(\mathbf{i}, \mathbf{j}, \mathbf{k})$. En pratique, étant donné le tétraèdre $\mathcal{T}$, il s'agit simplement d'appliquer la transformation (D.3) à tous les sommets

de $\mathcal{T}$ pour se ramener au repère $(\mathbf{u}, \mathbf{v}, \mathbf{w})$. Nous notons $\tilde{\mathcal{T}} = \{(u_i, v_i, w_i), i = 1, \dots, n\}$ l'image des sommets de $\mathcal{T}$ par $\mathbf{M}^{-1}(\boldsymbol{\theta})$ et $\tilde{\mathcal{A}}_{xy}(\tilde{\mathcal{T}})$ la projection de $\tilde{\mathcal{T}}$ sur le plan $w = \text{constante}$ suivant la direction $\mathbf{k}$. Alors nous avons $\forall(u, v), \mathcal{A}_{uv}(\mathcal{T}) = \tilde{\mathcal{A}}_{uv}(\tilde{\mathcal{T}})$.

Nous pouvons donc déduire que la dérivabilité de $\mathcal{A}_{uv}(\mathcal{T})$ par rapport à V_1 équivaut à celle de $\tilde{\mathcal{A}}_{xy}(\tilde{\mathcal{T}})$ par rapport à $\tilde{V}_1$. Si elles existent, leurs expressions analytiques sont reliées par :

$$\partial \mathcal{A}_{uv}(\mathcal{T}) / \partial V_1 = \partial \tilde{\mathcal{A}}_{uv}(\tilde{\mathcal{T}}) / \partial V_1 = \mathbf{M}(\boldsymbol{\theta}) \partial \tilde{\mathcal{A}}_{uv}(\tilde{\mathcal{T}}) / \partial \tilde{V}_1. \quad (\text{D.4})$$

Pour plus de simplicité, nous conservons les notations $\mathcal{T}$ et (x_i, y_i, z_i) pour désigner $\tilde{\mathcal{T}}$ et les coordonnées de ses sommets, et $\mathcal{A}_{xy}(\mathcal{T})$ pour désigner ses projections orthogonales sur le plan ($z = \text{constante}$). La position fixée dans le plan détecteur est dorénavant notée $(X, Y) = X \mathbf{i} + Y \mathbf{j}$.

D.3 Cas d'un tétraèdre

Dans tout ce paragraphe, nous notons $\{v_i = (x_i, y_i), i = 1, \dots, 4\}$ les coordonnées des projections des sommets V_i sur le plan $(\mathbf{i}, \mathbf{j})$. Trois configurations régulières peuvent survenir suivant les positions relatives des points v_i et celle du point v_1 (voir la figure D.1) :

- (1) v_1 se trouve à l'intérieur du triangle (v_2, v_3, v_4) .
- (2) L'un des sommets v_2, v_3 ou v_4 , disons v_2 , se trouve à l'intérieur du triangle formé par les trois autres sommets.
- (3) L'enveloppe convexe des quatre points est un quadrilatère. Dans ce dernier cas, nous procédons formellement à une renumérotation des sommets v_i de telle sorte que v_i et v_{i+1} sont des voisins dans le quadrilatère, et que les triangles (v_0, v_i, v_{i+1}) sont orientés dans le sens direct – en notant $v_5 = v_1$ et v_0 la position de l'intersection des diagonales du quadrilatère.

La projection est continue, à support polygonal (l'enveloppe convexe de (v_1, v_2, v_3, v_4) , donc un triangle ou un quadrilatère) et bi-affine par morceaux. Les morceaux sont 3 ou 4 sous-domaines triangulaires (voir la figure D.2 pour la représentation des projections dans les cas (1- 2) et (3)). Dans toute la suite, nous notons l la valeur du maximum de $p(x, y)$. La connaissance de la valeur de l , de la position (x, y) dont l'image par p est l et du support de la projection définit entièrement $p(x, y)$. Suivant les cas,

- $l = p(v_1)$, $l = p(v_2)$, et alors la valeur de p en tous les autres points v_i est nulle. Dans ce cas, le support de $p(x, y)$ est triangulaire et les « morceaux » sont 3 sous-triangles ;
- $p(v_i) = 0$ pour tout i , et $p(x, y)$ est maximale en un point v_0 , intersection des projections des diagonales du quadrilatère $\{v_i, i = 1, \dots, 4\}$, *i.e.*, (v_1, v_3) et (v_2, v_4) . Dans ce cas, les « morceaux » sont les 4 triangles (v_0, v_i, v_{i+1}) , $i = 1, \dots, 4$.

Nous procédons dans chaque cas à la détermination de l et à l'évaluation de $p(x, y)$ sur chacun des triangles sur lesquels p est bi-affine ; sur chacun de ces triangles, p prend la forme $p(x, y) = \alpha x + \beta y + \gamma$, où les constantes α, β et γ dépendent de la position du triangle.

Remarque Nous avons vu que pour un tétraèdre fixé, il existe trois cas de figure : (1), (2) et (3) quant à la configuration de sa projection. Nous attirons l'attention du lecteur sur le fait que ceci ne concerne qu'une direction fixée, et ne présage pas de la projection du même tétraèdre suivant une autre direction. La détermination du cas doit donc être faite pour chaque direction de projection.

Notations Dans tout ce qui suit, nous notons $\mathbf{f}_i$ la normale au côté du tétraèdre opposé au sommet V_i , orientée vers le plan détecteur, *i.e.*, vers les z croissants, $\boldsymbol{\theta}_i = (\theta_i, \varphi_i)$ ses angles sphériques associés ($0 \leq \varphi_i \leq \pi/2$), et $\hat{V}_i$ la projection de V_i parallèlement à $\mathbf{k}$ sur le plan formé par les autres sommets ; voir la figure D.3. Nous définissons également pour tout $(i, j, k, l) \in \{1, 4\}^4$,

$$\mathbf{P}_{ij/kl} = \begin{pmatrix} x_j - x_i & x_l - x_k \\ y_j - y_i & y_l - y_k \end{pmatrix} \quad \text{et} \quad \text{Det}(i, j/k, l) = \det \mathbf{P}_{ij/kl}. \quad (\text{D.5})$$

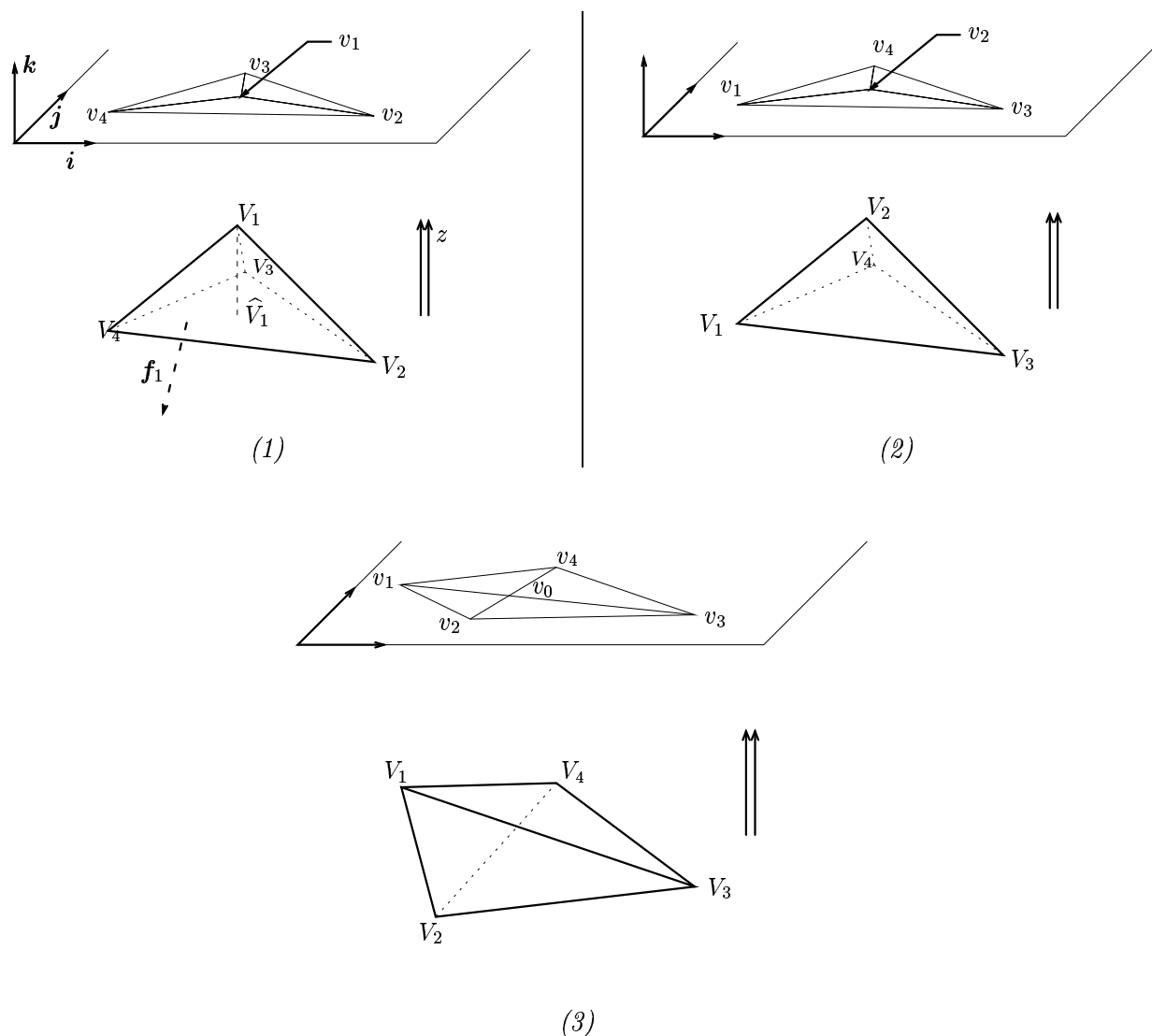


Figure D.1 – Support de la projection du tétraèdre $\mathcal{T}$ dans les trois cas (1), (2) et (3) respectivement.

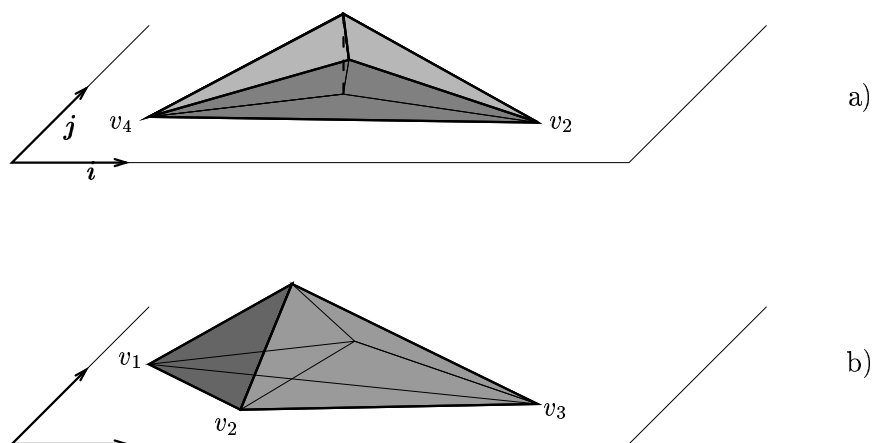


Figure D.2 – Représentation de la projection du tétraèdre $\mathcal{T}$ dans les configurations (1) et (2) – graphe (a), et dans la configuration (3) – graphe (b). Les projections sont affines par morceaux, où les morceaux correspondent aux trois ou quatre triangles qui constituent leur support.

D.3.1 Tétraèdre de type (1)

Évaluation de l et de sa dérivée partielle

Évaluer l revient à évaluer $\widehat{V}_1$. En notant $\varepsilon = 1$ si V_1 est au dessus du plan (V_2, V_3, V_4) , -1 sinon, nous avons :

- $\mathbf{V}_1 \widehat{V}_1 = -\varepsilon l \mathbf{k}$ car $l \geq 0$;
- $\widehat{V}_1 \in (V_2, V_3, V_4) \Rightarrow \langle \mathbf{V}_2 \widehat{V}_1, \mathbf{f}_1 \rangle = 0$.

En décomposant $\mathbf{V}_2 \widehat{V}_1$ en $\mathbf{V}_2 \mathbf{V}_1 + \mathbf{V}_1 \widehat{V}_1$, nous déduisons $\langle \mathbf{V}_2 \mathbf{V}_1, \mathbf{f}_1 \rangle - \varepsilon l \langle \mathbf{k}, \mathbf{f}_1 \rangle = 0$, soit :

$$l = \varepsilon \langle \mathbf{V}_2 \mathbf{V}_1, \mathbf{f}_1 \rangle / \cos \varphi_1. \quad (\text{D.6})$$

Ce quotient est bien licite, car $\cos \varphi_1 = 0$ signifierait que le plan (V_2, V_3, V_4) est parallèle à $\mathbf{k}$, c'est-à-dire que v_2, v_3 et v_4 sont alignés. Compte-tenu de l'hypothèse (1), v_1 serait également aligné avec tous les projetés v_i . Ce cas dégénéré est exclu, car il mène à des projections non différentiables.

Pour évaluer la dérivée partielle de l , nous remarquons que le terme $\varepsilon \mathbf{f}_1 / \cos \varphi_1$ est indépendant de V_1 . Nous déduisons aisément de (D.6) :

$$\partial l / \partial V_1 = \varepsilon \mathbf{f}_1 / \cos \varphi_1. \quad (\text{D.7})$$

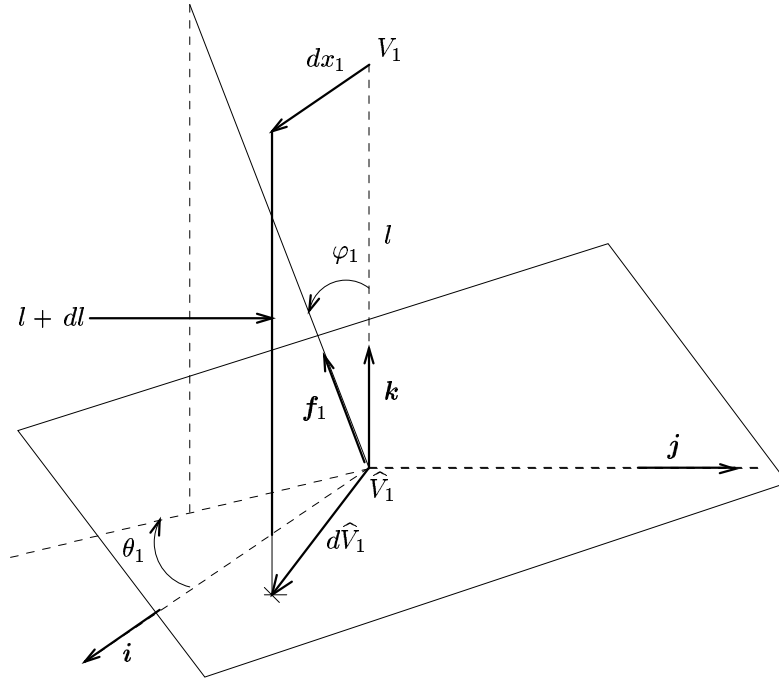


Figure D.3 – Déplacement du sommet V_1 dans la direction x . Le plan représenté est le support du triangle (V_2, V_3, V_4) . Le vecteur $\mathbf{f}_1$ est la normale unitaire à ce plan orientée vers V_1 , de coordonnées sphériques (θ_1, φ_1) relativement au repère $(\mathbf{i}, \mathbf{j}, \mathbf{k})$.

Évaluation de la dérivée partielle de $p(x, y)$ sur chaque sous-domaine

Lorsque V_1 est modifié, la position de v_1 l'est aussi, et nous avons $\partial v_1 / \partial x_1 = \mathbf{i}$, $\partial v_1 / \partial y_1 = \mathbf{j}$ et $\partial v_1 / \partial z_1 = \mathbf{0}$. Considérons le triangle de sommets v_1, v_i et v_{i+1} (en notant $v_5 = v_2$). Nous définissons les vecteurs $\mathbf{I} = \mathbf{v}_1 \mathbf{v}_i$ et $\mathbf{J} = \mathbf{v}_1 \mathbf{v}_{i+1}$. Dans le repère $(v_1, \mathbf{I}, \mathbf{J})$, nous notons $(\tilde{x}, \tilde{y})$ le système de coordonnées ; l'expression de p restreinte à (v_1, v_i, v_{i+1}) est $p(\tilde{x}, \tilde{y}) = l(1 - \tilde{x} - \tilde{y})$. La matrice de

passage de (i, j) à (I, J) est $P_{1,i/1,i+1}$, et nous avons :

$$(x, y)^t = (x_1, y_1)^t + P_{1,i/1,i+1} (\tilde{x}, \tilde{y})^t,$$

et

$$\begin{aligned} p(x, y) &= l \left(1 - P_{1,i/1,i+1}^{-1} (x - x_1, y - y_1)^t \cdot (i + j) \right) \\ &= l \left(1 - \frac{(y_{i+1} - y_i)(x - x_1) - (x_{i+1} - x_i)(y - y_1)}{\text{Det}(1, i/1, i + 1)} \right). \end{aligned} \quad (\text{D.8})$$

$p(x, y)$ est donc dérivable sur (v_1, v_i, v_{i+1}) , et nous avons :

$$\frac{\partial p(x, y)}{\partial V_1} = \frac{p(x, y)}{l} \frac{\partial l}{\partial V_1} + \frac{p(x, y)}{\text{Det}(1, i/1, i + 1)} \mathbf{V}_i \mathbf{V}_{i+1} \wedge \mathbf{k}. \quad (\text{D.9})$$

Dérivabilité de $p(x, y)$ sur $\mathbb{R}^2$

Les dérivées partielles par rapport à x_1 et y_1 de la restriction de $p(x, y)$ au i -ème domaine dépendent explicitement de x_i, y_i, x_{i+1} et y_{i+1} . Il n'y a donc pas (sauf cas particulier) de « raccordement » possible et les dérivées partielles de $p(x, y)$ par rapport à x_1 et y_1 ne peuvent pas être étendues aux segments (v_1, v_i) , malgré l'existence de dérivées partielles à droite et à gauche. En revanche, les dérivées partielles de $p(x, y)$ par rapport à z_1 ont une expression identique sur chacun des domaines, ε et l étant indépendants de i . Comme $p(x, y)$ est continue sur $\mathbb{R}^2$, $\partial p(x, y)/\partial z_1$ l'est aussi, ce qui garantit son extension sur les segments (v_1, v_i) . Enfin, sur les bords (v_i, v_{i+1}) , les expressions de toutes les dérivées partielles de $p(x, y)$, restreinte au triangle i , prennent la valeur nulle, ce qui garantit l'extension de ces dérivées en ces points.

En conclusion, $p(x, y)$ est dérivable sur $\mathbb{R}^2$ par rapport à z_1 . Ses dérivées par rapport à x_1 et y_1 sont définies partout sauf sur les segments (v_1, v_i) .

D.3.2 Tétraèdre de type (2)

Dans ce cas, l'expression de l est identique à celle du § D.3.1, dans laquelle les rôles de V_1 et V_2 sont intervertis. Cette expression

$$l = \varepsilon \langle \mathbf{V}_1 \mathbf{V}_2, \mathbf{f}_2 / \cos \varphi_2 \rangle, \quad (\text{D.10})$$

où $\varepsilon = 1$ si V_2 est au dessus du plan (V_1, V_3, V_4) , -1 sinon, est bien licite, car l'hypothèse $\cos \varphi_2 = 0$ combinée à l'hypothèse (2) mène à 4 projetés alignés. Ce cas singulier est exclu car il ne conduit pas à des projections différentiables.

En revanche, différencier l par rapport à V_1 est plus complexe qu'au § D.3.1 car un déplacement de V_1 modifie le plan sur lequel nous projetons V_2 . L'expression (D.10) dépend de la normale $\mathbf{f}_2$ au triangle (V_1, V_3, V_4) – plus précisément de $\mathbf{f}_2 / \cos \varphi_2$. Nous procédons donc au calcul de la dérivée partielle de $\mathbf{f}_2 / \cos \varphi_2$. Nous supposons, quitte à renuméroter V_3 et V_4 , que le triangle (V_1, V_3, V_4) est disposé de telle sorte que $\mathbf{V}_1 \mathbf{V}_3 \wedge \mathbf{V}_1 \mathbf{V}_4$ soit dirigé suivant les z croissants. Nous avons donc :

$$\mathbf{f}_2 = \frac{\mathbf{V}_1 \mathbf{V}_3 \wedge \mathbf{V}_3 \mathbf{V}_4}{\|\mathbf{V}_1 \mathbf{V}_3\| \|\mathbf{V}_3 \mathbf{V}_4\|} \quad \text{et} \quad \frac{\mathbf{f}_2}{\cos \varphi_2} = \frac{\mathbf{V}_1 \mathbf{V}_3 \wedge \mathbf{V}_3 \mathbf{V}_4}{\text{Det}(1, 3/3, 4)}. \quad (\text{D.11})$$

D'où

$$\begin{aligned} \partial (\mathbf{f}_2 / \cos \varphi_2) / \partial x_1 &= [\mathbf{i} \wedge \mathbf{V}_3 \mathbf{V}_4 + (y_4 - y_3) \mathbf{f}_2 / \cos \varphi_2] / \text{Det}(1, 3/3, 4), \\ \partial (\mathbf{f}_2 / \cos \varphi_2) / \partial y_1 &= -[\mathbf{j} \wedge \mathbf{V}_3 \mathbf{V}_4 + (x_4 - x_3) \mathbf{f}_2 / \cos \varphi_2] / \text{Det}(1, 3/3, 4), \\ \partial (\mathbf{f}_2 / \cos \varphi_2) / \partial z_1 &= -[\mathbf{k} \wedge \mathbf{V}_3 \mathbf{V}_4] / \text{Det}(1, 3/3, 4), \end{aligned}$$

et finalement :

$$\begin{aligned} \varepsilon \frac{\partial l}{\partial x_1} &= -\tan \varphi_2 \cos \theta_2 + \frac{(z_4 - z_3)(y_2 - y_1) + (y_4 - y_3)(\varepsilon l - z_2 + z_1)}{\text{Det}(1, 3/3, 4)}, \\ \varepsilon \frac{\partial l}{\partial y_1} &= -\tan \varphi_2 \sin \theta_2 - \frac{(z_4 - z_3)(x_2 - x_1) + (x_4 - x_3)(\varepsilon l - z_2 + z_1)}{\text{Det}(1, 3/3, 4)}, \\ \varepsilon \frac{\partial l}{\partial z_1} &= -\frac{\text{Det}(2, 3/3, 4)}{\text{Det}(1, 3/3, 4)}. \end{aligned} \quad (\text{D.12})$$

Nous obtenons de la même façon qu'au § D.3.1 les expressions des dérivées partielles de $p(x, y)$ dans chacun des triangles (V_2, V_1, V_3) , (V_2, V_1, V_4) et (V_2, V_3, V_4) :

— Dans le triangle (V_2, V_1, V_i) — $i = 3$ ou 4 , l'expression de $p(x, y)$ en fonction de l est identique à (D.8), où les indices 1 et 2 sont échangés :

$$p(x, y) = l \left(1 - \frac{(y_i - y_1)(x - x_2) - (x_i - x_1)(y - y_2)}{\text{Det}(2, 1/2, i)} \right). \quad (\text{D.13})$$

Nous en déduisons :

$$\frac{\partial p(x, y)}{\partial V_1} = \frac{p(x, y)}{l} \frac{\partial l}{\partial V_1} + \frac{[l \mathbf{V} \mathbf{V}_i - p(x, y) \mathbf{V}_2 \mathbf{V}_i] \wedge \mathbf{k}}{\text{Det}(2, 1/2, i)}, \quad (\text{D.14})$$

où $\mathbf{V}$ est le point $(x, y, 0)$.

— Dans le triangle (V_2, V_3, V_4) , nous avons de même :

$$p(x, y) = l \left(1 - \frac{(y_4 - y_3)(x - x_2) - (x_4 - x_3)(y - y_2)}{\text{Det}(2, 3/2, 4)} \right). \quad (\text{D.15})$$

et par conséquent, $\partial p(x, y)/\partial V_1 = p(x, y)/l \partial l/\partial V_1$.

Dérivabilité de $p(x, y)$ sur $\mathbb{R}^2$

— De même que dans le cas (1), $\partial p(x, y)/\partial z_1$ est définie de façon unique sur $\mathbb{R}^2$ par $\partial p(x, y)/\partial z_1 = p(x, y)/l \partial l/\partial z_1$.

— Les dérivées partielles par rapport à x_1 et y_1 ne sont pas définies sur les frontières (v_2, v_i) des sous-domaines triangulaires ($i \in \{1, 3, 4\}$), ni sur les bords (v_1, v_3) et (v_4, v_1) . En revanche, elles sont définies sur le bord (v_3, v_4) et y prennent la valeur nulle.

D.3.3 Tétraèdre de type (3)

Dans ce cas, $l = p(v_0)$. La difficulté réside sur le fait que le point v_0 n'est pas la projection d'un sommet du tétraèdre. Il faut donc, en plus du calcul de l , déterminer ses coordonnées.

Évaluation de l et de sa dérivée partielle

Le calcul de la valeur de l découle du volume du tétraèdre $\mathcal{T}$, qui s'exprime de deux façons. Il est donné par $\varepsilon \mathcal{V}(\mathcal{T})/6$, où :

$$\mathcal{V}(\mathcal{T}) = \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix} = \langle \mathbf{V}_4 \mathbf{V}_1, \mathbf{V}_2 \mathbf{V}_3 \wedge \mathbf{V}_2 \mathbf{V}_4 \rangle,$$

et $\varepsilon = \pm 1 = \text{sgn}(\mathcal{V}(\mathcal{T}))$ est choisi pour assurer la positivité du volume. Le volume s'écrit également :

$$\begin{aligned} \iint_{\mathbb{R}^3} f(x, y, z) \, dx \, dy \, dz &= \iint_{\mathbb{R}^2} \left(\int_{\mathbb{R}} f(x, y, z) \, dz \right) \, dx \, dy \\ &= \iint_{\mathbb{R}^2} p(x, y) \, dx \, dy = \sum_{i=1}^4 \mathcal{V}_i(l)/6, \end{aligned} \quad (\text{D.16})$$

où les termes $\mathcal{V}_i(l)/6$ sont les intégrales de $p(x, y)$ sur les triangles (v_0, v_i, v_{i+1}) . Compte tenu du caractère affine de la restriction de $p(x, y)$ à chaque triangle, chaque terme $\mathcal{V}_i(l)/6$ s'interprète comme le volume algébrique du tétraèdre de sommets $(x_0, y_0, 0)$, $(x_i, y_i, 0)$, $(x_{i+1}, y_{i+1}, 0)$ et (x_0, y_0, l) :

$$\mathcal{V}_i(l)/6 = \pm \begin{vmatrix} x_0 & y_0 & 0 & 1 \\ x_i & y_i & 0 & 1 \\ x_{i+1} & y_{i+1} & 0 & 1 \\ x_0 & y_0 & l & 1 \end{vmatrix} = l \begin{vmatrix} x_0 & y_0 & 1 \\ x_i & y_i & 1 \\ x_{i+1} & y_{i+1} & 1 \end{vmatrix} = l \, \text{Det}(0, i/0, i+1).$$

Le facteur ± 1 disparaît car l'orientation des triangles (v_0, v_i, v_{i+1}) dans le sens direct garantit la positivité des déterminants 3×3 . Finalement, l s'exprime analytiquement en fonction de $\{x_i, y_i, z_i, i = 1, \dots, 4\}$ par :

$$l = \frac{\varepsilon \mathcal{V}(\mathcal{T})}{\sum_{i=1}^4 \text{Det}(0, i/0, i+1)} = \frac{\varepsilon \mathcal{V}(\mathcal{T})}{\text{Det}(4, 1/4, 2) + \text{Det}(2, 3/2, 4)}, \quad (\text{D.17})$$

en remarquant que :

- chaque terme $\text{Det}(0, i/0, i+1)$ est le double de l'aire du triangle (v_0, v_i, v_{i+1}) .
- la somme de ces termes est le double de l'aire du quadrilatère $\{v_1, v_2, v_3, v_4\}$, soit le double de la somme des aires de (v_4, v_1, v_2) et (v_2, v_3, v_4) (voir la figure D.1).

Le terme $\text{Det}(2, 3/2, 4)$ est indépendant de la position du sommet V_1 , et les dérivées partielles de $\text{Det}(4, 1/4, 2)$ et de $\mathcal{V}(\mathcal{T})$ par rapport à V_1 s'expriment simplement par :

$$\partial \mathcal{V}(\mathcal{T}) / \partial V_1 = \mathbf{V}_2 \mathbf{V}_3 \wedge \mathbf{V}_2 \mathbf{V}_4 \quad \text{et} \quad \partial \text{Det}(4, 1/4, 2) / \partial V_1 = \mathbf{k} \wedge \mathbf{V}_2 \mathbf{V}_4. \quad (\text{D.18})$$

Nous déduisons de (D.17) l'expression analytique de la dérivée partielle de l :

$$\frac{\partial l}{\partial V_1} = \frac{[\varepsilon \mathbf{V}_2 \mathbf{V}_3 - l \mathbf{k}] \wedge \mathbf{V}_2 \mathbf{V}_4}{\text{Det}(4, 1/4, 2) + \text{Det}(2, 3/2, 4)}. \quad (\text{D.19})$$

Évaluation de la dérivée partielle de $p(x, y)$

Si le sommet v_0 n'intervient pas explicitement dans l'expression de l , ce n'est pas le cas pour $p(x, y)$. Nous procédons donc en plusieurs étapes :

- Calcul des coordonnées (x_0, y_0) de v_0 en fonction de celles des sommets de $\mathcal{T}$;
- Différentiation de v_0 par rapport à V_1 ;
- Calcul et différentiation de $p(x, y)$ par rapport à V_1 .

Le calcul de (x_0, y_0) s'obtient en résolvant un système linéaire. En tant qu'intersection des diagonales (v_1, v_3) et (v_2, v_4) du quadrilatère $\{v_i, i = 1, \dots, 4\}$, le couple (x_0, y_0) satisfait :

$$\text{Det}(0, 3/1, 3) = \text{Det}(0, 4/2, 4) = 0.$$

La résolution de ce système linéaire permet l'obtention de l'expression analytique de v_0 , en utilisant les formules de Cramer :

$$\mathbf{v}_0 = \left[(x_2 y_4 - x_4 y_2) \mathbf{v}_1 \mathbf{v}_3 - (x_1 y_3 - x_3 y_1) \mathbf{v}_2 \mathbf{v}_4 \right] / \text{Det}(1, 3/2, 4). \quad (\text{D.20})$$

Les dérivées partielles de x_0 et y_0 par rapport à x_1 et y_1 ont les expressions suivantes :

$$\begin{cases} \partial x_0 / \partial x_1 &= [(y_4 - y_3)(x_4 - x_2) - (x_4 - x_0)(y_4 - y_2)] / \text{Det}(1, 3/2, 4), \\ \partial y_0 / \partial x_1 &= -(y_3 - y_0)(y_4 - y_2) / \text{Det}(1, 3/2, 4), \\ \partial x_0 / \partial y_1 &= (x_3 - x_0)(x_4 - x_2) / \text{Det}(1, 3/2, 4), \\ \partial y_0 / \partial y_1 &= [(y_4 - y_0)(x_4 - x_2) - (x_4 - x_3)(y_4 - y_2)] / \text{Det}(1, 3/2, 4), \end{cases} \quad (\text{D.21})$$

et les dérivées partielles par rapport à z_1 sont nulles.

Dans chaque domaine triangulaire (v_0, v_i, v_{i+1}) , $i = 1, \dots, 4$, l'expression des projections est

$$p(x, y) = l \left(1 - \frac{(y_{i+1} - y_i)(x - x_0) - (x_{i+1} - x_i)(y - y_0)}{\text{Det}(0, i/0, i+1)} \right). \quad (\text{D.22})$$

— Le cas $i = 2$ ou 3 s'apparente au cas étudié au § D.3.1, car seul v_0 subit une modification suite à un déplacement de V_1 . Nous obtenons :

$$\partial \text{Det}(0, i/0, i+1) / \partial V_1 = -(y_{i+1} - y_i) \partial x_0 / \partial V_1 + (x_{i+1} - x_i) \partial y_0 / \partial V_1,$$

donc :

$$\frac{\partial p(x, y)}{\partial V_1} = \frac{p(x, y)}{l} \frac{\partial l}{\partial V_1} + \left[(y_{i+1} - y_i) \frac{\partial x_0}{\partial V_1} - (x_{i+1} - x_i) \frac{\partial y_0}{\partial V_1} \right] \frac{p(x, y)}{\text{Det}(0, i/0, i+1)}, \quad (\text{D.23})$$

et en particulier $\partial p(x, y) / \partial z_1 = \partial l / \partial z_1 p(x, y) / l$.

— Dans le cas du triangle (v_0, v_1, v_j) — $j = 2$ ou 4 , v_0 et v_1 sont modifiés lorsque V_1 est déplacé. Le calcul des dérivées partielles de $p(x, y)$ conduit à :

$$\partial \text{Det}(0, 1/0, j) / \partial V_1 = \mathbf{v}_0 \mathbf{v}_j \wedge \mathbf{k} - (y_j - y_1) \partial x_0 / \partial V_1 + (x_j - x_1) \partial y_0 / \partial V_1.$$

D'où :

$$\begin{aligned} \partial p(x, y) / \partial V_1 &= p(x, y) / l \partial l / \partial V_1 + \left[(l \mathbf{v} \mathbf{v}_j - p(x, y) \mathbf{v}_0 \mathbf{v}_j) \wedge \mathbf{k} + \right. \\ &\quad \left. p(x, y) \left((y_j - y_1) \partial x_0 / \partial V_1 - (x_j - x_1) \partial y_0 / \partial V_1 \right) \right] / \text{Det}(0, 1/0, j). \end{aligned} \quad (\text{D.24})$$

En particulier, $\partial p(x, y) / \partial z_1 = \partial l / \partial z_1 p(x, y) / l$.

Dérivabilité de $p(x, y)$ sur $\mathbb{R}^2$

Les conclusions sont identiques à celles du cas (2) :

— $\partial p(x, y) / \partial z_1$ est définie de façon unique sur $\mathbb{R}^2$ par $\partial p(x, y) / \partial z_1 = p(x, y) / l \partial l / \partial z_1$.

— Les dérivées partielles par rapport à x_1 et y_1 ne sont pas définies sur les frontières (v_0, v_i) des sous-domaines triangulaires ($i \in \{1, \dots, 4\}$), ni sur les bords (v_4, v_1) et (v_1, v_2) . En revanche, elles le sont sur les bords (v_2, v_3) et (v_3, v_4) et y prennent la valeur nulle.

D.3.4 Coût de la projection d'un tétraèdre et de sa dérivée

Nous présentons une évaluation du coût d'une projection $p(x, y)$ d'un tétraèdre $\mathcal{T}$, puis de sa dérivée par rapport à V_1 . Tout d'abord, rappelons que l'étape de changement de repère présentée au § D.2 et appliquée à l'ensemble des sommets du polyèdre $\mathbf{V}$ coûte $8n$ flops (8 multiplications par sommet). Quant à la recherche du type d'un tétraèdre $\mathcal{T} = (V_1, V_2, V_3, V_4)$, elle coûte 13 flops.

La fonction $p(x, y)$ étant affine par morceaux, c'est-à-dire de la forme $p(x, y) = \alpha_i x + \beta_i y + \gamma_i$ sur 3 ou 4 domaines triangulaires suivant le type de $\mathcal{T}$, son évaluation est extrêmement peu coûteuse. En effet, elle se réduit à l'évaluation des 9 ou 12 paramètres $\{\alpha_i, \beta_i, \gamma_i\}$, puis à $2\delta r$ opérations flottantes, en notant δr le nombre de pixels détecteurs dont le centre appartient au support de $p(x, y)$.

Quelquesoit le type de $\mathcal{T}$, la dérivée partielle de $p(x, y)$ par rapport à V_1 est toujours de la forme $p(x, y) [1/l \partial l / \partial V_1 + \boldsymbol{\alpha}_i] + \beta_i$, où les paramètres vectoriels $\{\boldsymbol{\alpha}_i, \beta_i\}$ dépendent directement des sommets de $\mathcal{T}$ ($\beta_i = \mathbf{0}$ dans les cas (1) et (3)). En supposant $p(x, y)$ déjà évaluée, le sur-coût permettant le calcul de $\partial p(x, y) / \partial V_1$ est donc de δr opérations flottantes, à une constante additive près.

D.4 Différentiabilité de $\mathcal{K}(\mathbf{V})$

D'après l'expression (V.26), la contribution d'un tétraèdre $\mathcal{T} = \mathcal{T}_r$ de la décomposition (V.21) à la k -ème somme $\partial\mathcal{K}_k/\partial V_1$ de la dérivée partielle $\partial\mathcal{K}/\partial V_1$ prend la forme :

$$2\varepsilon_1 \sum_{i,j=1}^{r_x, r_y} \frac{\partial \mathcal{A}_{ij}(\mathcal{T})}{\partial V_1} [\mathcal{A}_{ij}(\mathbf{V}) - d_{ij}]. \quad (\text{D.25})$$

Nous déduisons des études du § D.3 les résultats suivants quant à l'existence de (D.25), en fonction de la nature du tétraèdre $\mathcal{T}$ relativement à la projection k . Bien sûr, la différentiabilité de $\mathcal{K}(\mathbf{V})$ découle de l'existence des termes (D.25) d'après (V.26).

Proposition 10 *Quelquesoit le type de $\mathcal{T}$, la troisième composante de (D.25) est définie. Les dérivées partielles par rapport à x_1 et y_1 sont définies si et seulement si aucun pixel détecteur n'a son centre situé sur l'un des segments*

- (v_1, v_i) , $i = 2, 3, 4$ (cas (1)).
- (v_1, v_i) ou (v_2, v_i) , $i = 2, 3, 4$ (cas (2)).
- (v_0, v_i) , $i = 1, \dots, 4$ ou (v_1, v_2) ou (v_1, v_3) (cas (3)).

Remarque L'introduction d'un sommet fictif W_1 pour refermer le mini-polyèdre $\mathcal{P}_1$ peut avoir un effet non souhaité en introduisant des points de non différentiabilité supplémentaires dans la contribution de chaque tétraèdre extrait de $\mathcal{P}_1$ dans $\mathcal{K}$. Plus précisément, si l'on fixe un tétraèdre $\mathcal{T}$ de type (1) par exemple, les deux premières composantes de la quantité vectorielle (D.25) ne sont pas définies si le centre du pixel (i, j) du plan détecteur se trouve sur le segment (v_1, w_1) , bien que les dérivées partielles à droite et à gauche soient définies. En revanche, la somme des quantités (D.25) sur les n_1 tétraèdres qui forment $\mathcal{P}_1$ s'étend par continuité sur le segment (v_1, w_1) .

Numériquement, il se pose donc un problème concernant les éventuels pixels dont le centre est situé sur (v_1, w_1) . Quelle valeur attribuer aux quantités (D.25) en ces points ? C'est un problème que nous ne savons pas résoudre. Néanmoins, il est très peu probable que ce genre de situation se produise.

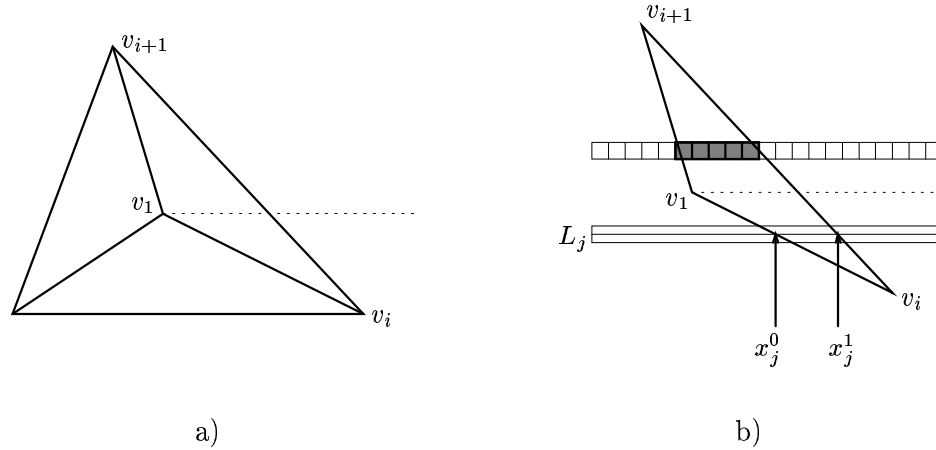


Figure D.4 – Évaluation de (D.25) dans le cas (1). a) Définition des trois sous-domaines triangulaires, b) Description de tous les pixels dont le centre se situe à l'intérieur du triangle (v_1, v_i, v_{i+1}) . En foncé, la liste des pixels parcourus pour une ligne fixée du sous-triangle supérieur, intersection de (v_1, v_i, v_{i+1}) et du demi-plan $y \geq y_1$ (sous-triangle repéré par $q = i + 1$).

Ordonner les sommets du triangle par ordonnées croissantes.
 Pour fixer les idées, nous supposons $y_i \leq y_1 \leq y_{i+1}$.
 Si la droite (v_i, v_{i+1}) est au dessus du point v_1 ,
 Pour $q \in \{i, i+1\}$,
 /* Parcours des sous-triangles inférieur ($q = i$) et supérieur ($q = i+1$) */
 Pour toute ligne L_j d'ordonnée (il s'agit de l'ordonnée commune du centre des pixels de cette ligne) comprise entre y_q et y_1 :
 1. Faire l'intersection de L_j avec (v_1, v_q) et $(v_i, v_{i+1}) \rightarrow$ positions x_j^0 et x_j^1 .
 2. Décrire tous les pixels de L_j dont le centre à une abscisse comprise dans l'intervalle (x_j^0, x_j^1) .
 Sinon,
 Pour $q \in \{i, i+1\}$,
 /* Parcours des sous-triangles inférieur ($q = i$) et supérieur ($q = i+1$) */
 Pour toute ligne L_j , d'ordonnée comprise entre y_q et y_1 :
 1. Faire l'intersection de L_j avec (v_i, v_{i+1}) et $(v_1, v_q) \rightarrow$ positions x_j^0 et x_j^1 .
 2. Décrire tous les pixels de L_j dont le centre à une abscisse comprise dans l'intervalle (x_j^0, x_j^1) .

Tableau D.1 – Algorithme de parcours d'un triangle pour l'évaluation de (D.25).

D.5 Évaluation de $\nabla \mathcal{K}(\mathbf{V})$

Nous dérivons de la proposition 10 un algorithme d'évaluation de (D.25), la dérivée partielle $\partial \mathcal{K} / \partial V_1$ étant la somme de ces termes sur toutes les projections k et tous les tétraèdres $\mathcal{T}$ de la décomposition (V.21). Sans perte de généralité, nous nous plaçons dans le cas où $\mathcal{T}$ est de type (1), les autres cas étant similaires. En supposant que l'on soit dans les conditions d'existence de (D.25), son évaluation nécessite de décrire les trois sous-triangles (v_1, v_i, v_{i+1}) . Pour chacun, nous effectuons :
 — le parcours de tous les pixels détecteurs dont le centre est contenu dans (v_1, v_i, v_{i+1}) ;
 — le calcul, pour chacun de ces pixels, des expressions de $p(x, y)$ et de ses dérivées partielles en substituant à (x, y) les coordonnées de son centre.

Le triangle (v_1, v_i, v_{i+1}) , illustré en figure D.4, est subdivisé en deux sous-triangles. Il est parcouru suivant l'algorithme décrit dans le tableau D.1.

D.6 Cas où les directions ne sont pas orthogonales au détecteur

Le principe du calcul ne change pas. Pour une projection fixée, on définit $\mathbf{n}$, normale au plan détecteur orientée par la direction de projection ; en notant $\mathbf{w}$ le vecteur dirigeant la projection, nous choisissons $\mathbf{n}$ tel que $\langle \mathbf{n}, \mathbf{w} \rangle \geq 0$. On se ramène à la projection sur le plan $z = Z$ en effectuant comme au § D.2 une rotation qui transforme $\mathbf{n}$ en $\mathbf{k}$. Notons à présent $\mathbf{c} = \mathbf{c}_\theta$ l'image du vecteur $\mathbf{w}$ et $V_i = (x_i, y_i, z_i)$ les images des sommets de $\mathcal{P}$ par cette rotation. Par exemple, le cas où les directions sont orthogonales au plan détecteur mène à $\mathbf{c} = \mathbf{k}$. Tout point $V = (x, y, z) \in \mathbb{R}^3$ se projette suivant $\mathbf{c}$ sur le plan ($z = Z$) en $\hat{V} = (\hat{x}, \hat{y}, Z)$, avec :

$$\begin{aligned}\hat{x} &= x - (z - Z) \cos \theta \tan \varphi, \\ \hat{y} &= y - (z - Z) \sin \theta \tan \varphi.\end{aligned}\tag{D.26}$$

Par la suite, nous profitons des résultats du § D.3 pour étudier la dérivée partielle par rapport à V_1 de la projection d'un tétraèdre $\mathcal{T} = \{V_1, V_2, V_3, V_4\}$ suivant $\mathbf{c}$ sur le plan $z = Z$. De même que dans le cas « plan détecteur orthogonal », la projection d'un tétraèdre est à support polygonal (triangle ou quadrilatère). Ce support est défini comme l'enveloppe convexe des projections $\widehat{V}_i$ des sommets de $\mathcal{T}$. De plus, la projection est également bi-affine par morceaux, et nulle sur la frontière du support (dans les cas réguliers). Elle est donc définie par la position de son maximum et sa valeur l .

Évaluation de l et de sa dérivée partielle

Dans le cas « orthogonal » comme dans le cas « non orthogonal », nous projetons $\mathcal{T}$ suivant la même direction $\mathbf{c}$. Les valeurs de l sont donc identiques. En d'autres termes, il suffit d'effectuer une rotation qui transforme $\mathbf{c}$ en $\mathbf{k}$, de transformer tous les sommets V_i de $\mathcal{T}$ en V_i^r par cette rotation et d'étudier la projection orthogonale du tétraèdre $\mathcal{T}^r$ sur le plan $z = Z$ pour déduire la valeur de l . Nous choisissons la rotation de matrice $\mathbf{M}(\boldsymbol{\theta})$, définie en (D.3). Les transformés $V_i^r = (x_i^r, y_i^r, z_i^r)$ des sommets V_i sont définis par :

$$V_i^r = \mathbf{M}^{-1}(\boldsymbol{\theta}) V_i = \mathbf{M}^t(\boldsymbol{\theta}) V_i,$$

en utilisant l'orthogonalité de la matrice $\mathbf{M}(\boldsymbol{\theta})$.

— Dans les cas (1) et (2), la relation (D.6) mène aux expressions de l en fonction de V_1^r, V_2^r et des normales $\mathbf{f}_1^r$ et $\mathbf{f}_2^r$ aux plans (V_2^r, V_3^r, V_4^r) et (V_1^r, V_3^r, V_4^r) , dirigées vers les z croissants. Et les expressions (D.7) et (D.12) donnent la dérivée de l par rapport à V_1^r . Pour obtenir $\partial l / \partial V_1$, nous appliquons la formule de changement de variables :

$$\partial l / \partial V_1 = \mathbf{M}(\boldsymbol{\theta}) \partial l / \partial V_1^r.$$

Par la suite, nous nous intéressons à la dérivée partielle de l par rapport à $\widehat{V}_1$ car la projection $p(x, y)$ s'exprime en fonction des projections $v_i = (\hat{x}_i, \hat{y}_i)$ des sommets V_i suivant $\mathbf{c}$ sur le plan $z = Z$. La relation (D.26) implique :

$$\begin{pmatrix} \hat{x}_1 \\ \hat{y}_1 \\ \hat{z}_1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -\cos \theta \tan \varphi \\ 0 & 1 & -\sin \theta \tan \varphi \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + Z \begin{pmatrix} \cos \theta \tan \varphi \\ \sin \theta \tan \varphi \\ 0 \end{pmatrix},$$

soit $\widehat{V}_1 = \mathbf{N}(\boldsymbol{\theta}) V_1 + \mathbf{C}(\boldsymbol{\theta})$ en notation compacte. Nous en déduisons :

$$\partial l / \partial \widehat{V}_1 = \mathbf{N}(\boldsymbol{\theta})^{-t} \partial l / \partial V_1 = \mathbf{N}(\boldsymbol{\theta})^{-t} \mathbf{M}(\boldsymbol{\theta}) \partial l / \partial V_1^r,$$

où la transposée de l'inverse de $\mathbf{N}(\boldsymbol{\theta})$ est :

$$\mathbf{N}(\boldsymbol{\theta})^{-t} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \cos \theta \tan \varphi & \sin \theta \tan \varphi & 1 \end{pmatrix},$$

et

$$\mathbf{N}(\boldsymbol{\theta})^{-t} \mathbf{M}(\boldsymbol{\theta}) = \begin{pmatrix} -\cos \theta \cos \varphi & \sin \theta & \cos \theta \sin \varphi \\ -\sin \theta \cos \varphi & -\cos \theta & \sin \theta \sin \varphi \\ 0 & 0 & 1 / \cos \varphi \end{pmatrix}.$$

— Dans le cas (3), l'expression (D.17) relie l au volume du tétraèdre $\mathcal{T}^r = (V_1^r, V_2^r, V_3^r, V_4^r)$ — i.e., celui de $\mathcal{T}$, et aux projections $v_i^r = (x_i^r, y_i^r)$ des sommets de $\mathcal{T}^r$ suivant $\mathbf{k}$ sur le plan $z = Z$:

$$l = \varepsilon \mathcal{V}(\mathcal{T}^r) / [Det(4, 1/4, 2) + Det(2, 3/2, 4)], \quad (\text{D.27})$$

où les coordonnées (x_i^r, y_i^r) sont prises en compte dans les déterminants. Le résultat (D.19) est réutilisable et donne l'expression de la dérivée partielle de l par rapport à V_1^r :

$$\frac{\partial l}{\partial V_1^r} = \frac{[\varepsilon \mathbf{V}_2^r \mathbf{V}_3^r - l \mathbf{k}] \wedge \mathbf{V}_2^r \mathbf{V}_4^r}{\text{Det}(4, 1/4, 2) + \text{Det}(2, 3/2, 4)}.$$

De la même façon que dans les cas (1) et (2), nous effectuons le changement de variables qui permet de déduire la dérivée partielle de l par rapport à $\widehat{V}_1$.

Évaluation de $p(x, y)$ et de sa dérivée partielle

Que l'on projette sur un plan orthogonal à $\mathbf{c}$ ou non, deux cas émergent suivant que l'un des sommets a une projection non nulle (cas (1) ou (2), la projection a un support triangulaire) ou pas (cas (3), la projection a un support quadrangulaire). Or nous avons vu au paragraphe précédent que la projection d'un sommet suivant une direction fixée est indépendante du plan sur lequel nous projetons. Plus précisément, s'il existe un sommet V_{i_0} dont la projection sur $z = Z$ suivant $\mathbf{c}$ n'est pas nulle, alors la projection orthogonale du même point $V_{i_0}^r$ suivant $\mathbf{c}$ est également non nulle. Donc les cas (1), (2) et (3) coïncident pour une projection suivant $\mathbf{c}$, qu'elle soit orthogonale ou non. De plus, comme les repères $(\mathbf{i}, \mathbf{j})$ et $(\mathbf{u}, \mathbf{v})$ sont tous deux directs, l'ordre trigonométrique des 3 ou 4 sommets extérieurs est identique dans les deux cas. Cet ordre est important car il conditionne l'écriture des projections du tétraèdre. Cet ordre peut être directement obtenu lors de l'évaluation de l , ce qui évite les calculs superflus.

Les expressions obtenues au § D.3 donnent $p(x, y)/l$ en fonction des variables $\hat{x}_i$ et $\hat{y}_i$, les projections des sommets V_i sur le plan $z = Z$ suivant la direction $\mathbf{c}$. Par exemple, l'expression (D.8) se transforme en

$$p(x, y) = l \left(1 - \frac{(\hat{y}_3 - \hat{y}_2)(x - \hat{x}_1) - (\hat{x}_3 - \hat{x}_2)(y - \hat{y}_1)}{\text{Det}(1, 2/1, 3)} \right),$$

où $\text{Det}(1, 2/1, 3) = (\hat{x}_2 - \hat{x}_1)(\hat{y}_3 - \hat{y}_1) - (\hat{x}_3 - \hat{x}_1)(\hat{y}_2 - \hat{y}_1)$.

— Cas (1) et (2) : les expressions (D.9) et (D.14) de la dérivée partielle de $p(x, y)$ en fonction de l sont réutilisables et relient ainsi $\partial p(x, y)/\partial \widehat{V}_1$ à $\partial l/\partial \widehat{V}_1$. Par exemple, (D.9) devient :

$$\frac{\partial p(x, y)}{\partial \widehat{V}_1} = \frac{p(x, y)}{l} \frac{\partial l}{\partial \widehat{V}_1} + \frac{p(x, y)}{\text{Det}(1, 2/1, 3)} \mathbf{V}_2^r \mathbf{V}_3^r \wedge \mathbf{k}.$$

— Cas (3) : l'expression (D.22) se réécrit :

$$p(x, y) = l \left(1 - \frac{(\hat{y}_{i+1} - \hat{y}_i)(x - \hat{x}_0) - (\hat{x}_{i+1} - \hat{x}_i)(y - \hat{y}_0)}{\text{Det}(0, i/0, i+1)} \right),$$

où $\text{Det}(0, i/0, i+1) = (\hat{x}_i - \hat{x}_0)(\hat{y}_{i+1} - \hat{y}_0) - (\hat{x}_{i+1} - \hat{x}_0)(\hat{y}_i - \hat{y}_0)$. Ces expressions font intervenir le point $\hat{v}_0 = (\hat{x}_0, \hat{y}_0)$ défini de même que dans le cas « orthogonal » comme l'intersection des diagonales du quadrilatère $\{\hat{v}_i, i = 1, \dots, 4\}$. L'expression de $\hat{v}_0$ est identique à (D.20), où les variables x_i et y_i , $i = 1, \dots, 4$ sont remplacées par $\hat{x}_i$ et $\hat{y}_i$. De la même façon nous déduisons de (D.21) les dérivées partielles de $\hat{x}_0$ et $\hat{y}_0$ par rapport à $\hat{x}_1$ et $\hat{y}_1$. Et les expressions (D.23) et (D.25) nous mènent à celles de $p(x, y)$ par rapport aux coordonnées de $\widehat{V}_1$. Par exemple, (D.25) se généralise en :

$$\frac{\partial p(x, y)}{\partial \widehat{V}_1} = \frac{p(x, y)}{l} \frac{\partial l}{\partial \widehat{V}_1} + \left[(\hat{y}_{i+1} - \hat{y}_i) \frac{\partial \hat{x}_0}{\partial \widehat{V}_1} - (\hat{x}_{i+1} - \hat{x}_i) \frac{\partial \hat{y}_0}{\partial \widehat{V}_1} \right] \frac{p(x, y)}{\text{Det}(0, i/0, i+1)}.$$

Finalement, dans ce cas comme dans les cas (1) et (2), un changement de coordonnées permet l'obtention de $\partial p(x, y)/\partial V_1$ via la relation :

$$\partial p(x, y)/\partial V_1 = \mathbf{N}(\boldsymbol{\theta})^t \partial p(x, y)/\partial \widehat{V}_1, \quad (\text{D.28})$$

et la discussion sur les positions de non différentiabilité de $p(x, y)$ par rapport à x_1 et y_1 est donc induite par celles du § D.3.

ANNEXE E

ANGLES SOLIDES D'UN POLYÈDRE

Cette annexe est un complément à la section V.2.3 qui concerne la régularisation du modèle polyédrique à partir des angles solides $A_i(\mathbf{V})$ en ses sommets. Le terme de régularisation $\mathcal{R}(\mathbf{V}) = \sum_{i=1}^n \phi(\kappa_i(\mathbf{V}))$ est défini sous forme séparable à partir des courbures $\kappa_i(\mathbf{V}) = 1 + \cos(A_i(\mathbf{V})/2)$ en les sommets de $\mathbf{V}$. Dans le § V.2.3, nous avons vu que le calcul de l'angle solide en un sommet de $\mathbf{V}$, par exemple V_1 , se ramène à n_1 calculs d'angles solides de tétraèdres ayant V_1 pour sommet, où n_1 est le nombre de voisins de V_1 . De même que pour le calcul de $\nabla \mathcal{K}$, la tétraédrisation s'appuie sur une direction « moyenne » fictive :

$$\boldsymbol{\tau}_1 = \frac{1}{n_1} \left(\sum_{i=1}^{n_1} \boldsymbol{\tau}_{1i} \right), \quad (\text{E.1})$$

ou d'un point fictif W_1 sur la demi-droite $(V_1, \boldsymbol{\tau}_1)$. Et nous avons

$$A_1(\mathbf{V}) = \sum_{i=1}^{n_1} A_1(V_1, V_{1 \rightarrow i}, V_{1 \rightarrow i+1}, W_1). \quad (\text{E.2})$$

L'étude de $A_1(\mathbf{V})$ et de ses dérivées partielles se ramène donc au cas d'un tétraèdre.

Dans toute cette annexe, nous considérons donc un tétraèdre $\mathcal{T} = \{V_i, i = 1, \dots, 4\}$ et exprimons $A_1(\mathcal{T})$, noté A_1 , et ses dérivées partielles par rapport à V_1 et un autre sommet, disons V_2 . Nous déduisons au § E.4 un algorithme de calcul de $\mathcal{R}(\mathbf{V})$ et de son gradient.

E.1 Angles solides d'un tétraèdre

Nous définissons les notions relatives au tétraèdre $\mathcal{T}$ et illustrées en figure V.1.

- les vecteurs *unitaires* $\boldsymbol{\tau}_i$, $i = 2, 3, 4$, des arêtes adjacentes à V_1 ;
- les vecteurs $\mathbf{a}_i = \mathbf{V}_1 \mathbf{V}_i$, $i = 2, 3, 4$, et $a_i = \|\mathbf{a}_i\|$. Ainsi, $\mathbf{a}_i = a_i \boldsymbol{\tau}_i$;
- les projetés $\tilde{V}_i = V_1 + \boldsymbol{\tau}_i$ de V_i sur la sphère unité centrée en V_1 ; et
- le tétraèdre unitaire $\tilde{\mathcal{T}} = (V_1, \tilde{V}_2, \tilde{V}_3, \tilde{V}_4)$ associé à V_1 .

L'angle solide $A_1(\mathcal{T})$ est défini par la surface du « triangle sphérique » formé des sommets $\tilde{V}_i$. Cet angle, mesuré en stéradians, appartient à l'intervalle $(0, 2\pi)$, et en outre, en notant A_i les angles solides en les autres sommets de $\mathcal{T}$, nous avons d'après (Liu et Joe, 1994) :

$$0 \leq \sum_{i=1}^4 A_i \leq 2\pi. \quad (\text{E.3})$$

En particulier, si l'un des angles solides de $\mathcal{T}$ est important, alors tous les autres sont faibles et le tétraèdre possède 3 sommets singuliers (*i.e.*, à forte courbure).

Dans la littérature, les angles solides ne sont pas calculés directement, mais par l'intermédiaire de fonctions trigonométriques. En effet, il existe des expressions simples de $\sin A_1/2$ et $\tan A_1/2$. Comme $A_1 \in (0, 2\pi)$, $\sin A_1/2$ ne spécifie pas totalement A_1 , à la différence de $t(A_1) = \tan A_1/2$. Eriksson (1990) établit l'expression de $t(A_1)$ en fonction du tétraèdre $\tilde{\mathcal{T}}$:

$$t(A_1) = \frac{\varepsilon \mathcal{V}(\tilde{\mathcal{T}})}{1 + \tau_2 \tau_3 + \tau_2 \tau_4 + \tau_3 \tau_4}, \quad (\text{E.4})$$

où $\mathcal{V}(\tilde{\mathcal{T}}) = [\tau_2, \tau_3, \tau_4]$ est, à un coefficient près, le volume algébrique de $\tilde{\mathcal{T}}$ déjà défini au § D.3.3, et $\varepsilon = \text{sgn}(\mathcal{V}(\tilde{\mathcal{T}}))$. Nous en déduisons facilement l'expression de $t(A_1)$ en fonction des arêtes de $\mathcal{T}$:

$$t(A_1) = \frac{\varepsilon \mathcal{V}(\mathcal{T})}{a_2 a_3 a_4 + a_2 \mathbf{a}_3 \mathbf{a}_4 + a_3 \mathbf{a}_2 \mathbf{a}_4 + a_4 \mathbf{a}_2 \mathbf{a}_3}. \quad (\text{E.5})$$

Cette expression est relativement simple, car elle ne fait intervenir que le volume de $\mathcal{T}$ et les arêtes adjacentes à V_1 . En revanche, il existe des sommets réguliers, dits « orthogonaux », pour lesquels (E.5) n'est pas définie. Pour cette raison, nous préférons utiliser la fonction $\text{ct}(A_1) = \tan^{-1} A_1/2$ qui est monotone et finie pour tout sommet régulier.

Dans les calculs qui suivent, nous aurons besoin de $\cos A_1/2$. Un calcul direct de $\cos A_1/2$ à partir de $t(A_1)$ est relativement coûteux car nécessite un appel aux fonctions $\cos$ et $\arctan$. Nous préférons utiliser l'expression de $\sin A_1/2$ démontrée par Liu et Joe (1994) :

$$\sin A_1/2 = 2 \varepsilon \mathcal{V}(\mathcal{T}) \left[\prod_{2 \leq i < j \leq 4} (a_i + a_j + a_{ij}) (a_i + a_j - a_{ij}) \right]^{-1/2}, \quad (\text{E.6})$$

couplée avec l'équation (E.5). Dans l'expression (E.6), légèrement plus complexe que (E.5) mais définie pour tout tétraèdre dont les faces ne sont pas dégénérées ((E.6) est en particulier définie pour les tétraèdres plans), la notation a_{ij} représente la distance $d(V_i, V_j)$.

E.2 Différenciation des angles solides pour un tétraèdre non singulier

Nous entendons par « tétraèdre non singulier » les tétraèdres à volumes non nuls, *i.e.*, dont les sommets ne sont pas coplanaires. La fonction $\text{ct}(A_1)$ est donc finie, et nous l'utilisons pour le calcul des dérivées de $A_1(\mathcal{T})$ par rapport à V_1 et V_2 ; nous commençons par évaluer $\partial \text{ct}(A_1)/\partial$, puis nous déduisons $\partial A_1/\partial = -2 \sin^2(A_1/2) \partial \text{ct}(A_1)/\partial$.

— Le calcul de la dérivée par rapport à V_2 est relativement simple; a_3 et a_4 sont indépendants de V_2 , et $\partial a_2/\partial V_2 = \mathbf{a}_2/a_2 = \tau_2$, donc la dérivée du dénominateur de (E.5) est

$$\begin{aligned} \xi_2(\mathcal{T}) &= (a_3 a_4 + \mathbf{a}_3 \mathbf{a}_4) \tau_2 + a_3 \mathbf{a}_4 + a_4 \mathbf{a}_3 \\ &= a_3 a_4 \left[\sum_{i=2}^4 \tau_i \right] + (\mathbf{a}_3 \mathbf{a}_4) \tau_2. \end{aligned} \quad (\text{E.7})$$

— De même, $\partial a_i/\partial V_1 = -\tau_i$, d'où la dérivée du même terme par rapport à V_1 :

$$\begin{aligned} \xi_1(\mathcal{T}) &= - \sum_{i=2}^4 [(\mathbf{a}_j \mathbf{a}_k + a_j a_k) \tau_i + (a_j + a_k) \mathbf{a}_i] \\ &= - \left(\sum_{i < j} a_i a_j \right) \left[\sum_{i=2}^4 \tau_i \right] - \sum_{i=2}^4 (\mathbf{a}_j \mathbf{a}_k) \tau_i, \end{aligned} \quad (\text{E.8})$$

où j et k désignent systématiquement les deux indices de $\{2, 3, 4\}$ distincts de i . Finalement, les dérivées de $\text{ct}(A_1)$ s'écrivent :

$$\partial \text{ct}(A_1)/\partial V_k = \left[\varepsilon \xi_k(\mathcal{T}) - \text{ct}(A_1) \partial \mathcal{V}(\mathcal{T})/\partial V_k \right] / \mathcal{V}(\mathcal{T}), \quad (\text{E.9})$$

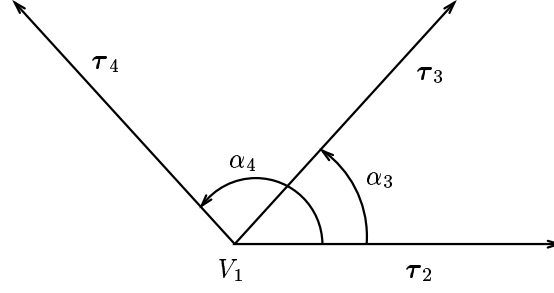


Figure E.1 – Tétraèdre plan. Cas où les directions adjacentes à V_1 sont situées dans le même demi-plan. L'angle solide vaut $A_1(\mathcal{T}) = 0$.

d'où celles de A_1 :

$$\partial A_1 / \partial V_k = 2 \sin^2(A_1/2) \left[\text{ct}(A_1) \partial \mathcal{V}(\mathcal{T}) / \partial V_k - \varepsilon \xi_k(\mathcal{T}) \right] / \mathcal{V}(\mathcal{T}), \quad (\text{E.10})$$

et en particulier $\partial A_1 / \partial V_k = -2 \xi_k(\mathcal{T}) / |\mathcal{V}(\mathcal{T})|$ si $A_1 = \pi$. Le calcul (E.10) fait intervenir l'équation (E.6) pour l'évaluation de $\sin(A_1/2)$.

E.3 Différenciation des angles solides pour un tétraèdre singulier

Nous étudions ici le cas où $\mathcal{V}(\mathcal{T}) = 0$, *i.e.*, les sommets de $\mathcal{T}$ sont coplanaires. D'après (E.5), $t(A_1) = 0$, donc $A_1 = 0$ ou 2π . Nous ne considérons pas, pour l'instant, le cas dégénéré où deux arêtes du tétraèdre ont des directions confondues. Nous énonçons la proposition suivante quant à la valeur de A_1 :

Proposition 11 *L'angle solide A_1 est nul si et seulement si V_1 appartient au triangle (V_2, V_3, V_4) .*

Preuve 10 Sans perte de généralité, nous supposons que le plan qui contient $\mathcal{T}$ est parallèle à (xy) , que $\tau_2 = \mathbf{i}$ et τ_3 est orienté vers les y positifs. Nous notons alors $\tau_3 = \mathbf{w}_{\alpha_3}$ et $\tau_4 = \mathbf{w}_{\alpha_4}$, où $\alpha_3 \in]0, \pi[$ et $\alpha_4 \in (0, 2\pi)$ (cf. figure E.1). L'expression de $\cos A_1/2$ que l'on obtient en divisant (E.6) par (E.5) est de même signe que

$$C(\alpha_4) = 1 + \cos \alpha_3 + \cos \alpha_4 + \cos(\alpha_4 - \alpha_3), \quad (\text{E.11})$$

donc

$$A_1 = 0 \iff C(\alpha_4) > 0. \quad (\text{E.12})$$

Nous étudions donc le signe de C sur $(0, 2\pi)$.

— Sur l'intervalle (α_3, π) , C est décroissante car $\alpha_4 - \alpha_3 \in (0, \pi)$. Comme $C(\pi) = 0$, C prend des valeurs positives sur cet intervalle.

— C est dérivable sur $(0, 2\pi)$, et $C'(\alpha_4) = -\sin \alpha_4 - \sin(\alpha_4 - \alpha_3)$. Sur $(0, \alpha_3)$, $C'(\alpha_4) = 0 \iff \sin(\alpha_3 - \alpha_4) = \sin \alpha_4 \iff \alpha_4 = \alpha_3/2$, le cas $\alpha_3 = \pi$ étant exclus par hypothèse. Comme C' est continue et $C'(0) = -C'(\alpha_3) = \sin \alpha_3 > 0$, C est croissante sur $(0, \alpha_3/2)$ et décroissante sur $(\alpha_3/2, \alpha_3)$. Enfin, comme $C(0) = C(\alpha_3) = 2 + 2 \cos \alpha_3 > 0$, C est positive sur $(0, \alpha_3)$.

— Enfin, pour tout α_4 , nous avons $C(\alpha_4) = 2(1 + \cos \alpha_3) - C(\alpha_4 - \pi)$. Par conséquent, C est décroissante sur $(\pi, \pi + \alpha_3/2)$ puis croissante sur $(\pi + \alpha_3/2, 2\pi)$. Comme $C(\pi) = C(\pi + \alpha_3) = 0$, nous déduisons que C est négative sur $(\pi, \pi + \alpha_3)$ et positive sur $(\pi + \alpha_3, 2\pi)$.

Nous déduisons de cette étude et de l'équivalence (E.12) que $A_1 = 0 \iff \alpha_4 \in (0, \pi)$ ou $\alpha_4 > \alpha_3 + \pi$, c'est-à-dire que $A_1 = 0$ si les vecteurs τ_2, τ_3, τ_4 sont situés dans le même demi-plan. $\square$

Comme $\tan(. / 2)$ est définie et continue en 0 et 2π , de dérivées égales à $1/2$, A_1 est dérivable par rapport à V_k si et seulement si $t(A_1)$ l'est, et dans ce cas, $\partial A_1 / \partial V_k = 2 \partial t(A_1) / \partial V_k$. Il est clair que $\mathcal{V}(\mathcal{T})$ est dérivable par rapport à V_k (voir § D.3.3 pour le calcul de sa dérivée). Comme $\mathcal{V}(\mathcal{T})$ s'annule au voisinage de V_k , le terme $\varepsilon = \pm 1$ de l'équation (E.5) peut changer de valeur quand le sommet V_k est déplacé de part et d'autre du plan de $\mathcal{T}$. Nous définissons donc pour tout k , les vecteurs $\varepsilon_{k,d}$ et $\varepsilon_{k,g} \in \{-1, 1\}^3$ dont les composantes sont égales aux signes de $\mathcal{V}(\mathcal{T} + d\mathcal{T})$ lorsque V_k est déplacé dans chaque direction i, j, k à droite et à gauche respectivement. $t(A_1)$ possède donc 2 dérivées (vectorielles) à droite et à gauche de V_k égales à :

$$\partial A_1 / \partial V_{k,.} = \frac{2 \varepsilon_{k,.} \partial \mathcal{V}(\mathcal{T}) / \partial V_k}{a_2 a_3 a_4 + a_2 \mathbf{a}_3 \mathbf{a}_4 + a_3 \mathbf{a}_2 \mathbf{a}_4 + a_4 \mathbf{a}_2 \mathbf{a}_3}, \quad (\text{E.13})$$

où le produit au numérateur est un produit composante par composante. Or, $A_1 = 0$ ou 2π et doit rester dans l'intervalle $(0, 2\pi)$. En d'autres termes, les vecteurs $\varepsilon_{k,g}$ et $\varepsilon_{k,d}$ sont forcément égaux. De plus, nous avons vu (cf. preuve 10) que $A_1 = 0$ si et seulement si le dénominateur de (E.13) est positif. Par conséquent, $\partial A_1 / \partial V_k$ et ce dénominateur doivent être de même signe. Donc $\varepsilon_{k,d} = \text{sgn}(\partial \mathcal{V}(\mathcal{T}) / \partial V_k)$, et l'équation (E.13) se récrit :

$$\partial A_1 / \partial V_k = \frac{2 |\partial \mathcal{V}(\mathcal{T}) / \partial V_k|}{a_2 a_3 a_4 + a_2 \mathbf{a}_3 \mathbf{a}_4 + a_3 \mathbf{a}_2 \mathbf{a}_4 + a_4 \mathbf{a}_2 \mathbf{a}_3}. \quad (\text{E.14})$$

Enfin, dans le cas dégénéré où deux des arêtes adjacentes à V_1 , disons (V_1, V_2) et (V_1, V_4) , ont des directions confondues, A_1 ne peut être défini que si ces directions sont orientées dans le même sens, i.e., $\tau_2 = \tau_4$. En effet, dans le cas contraire, le dénominateur de (E.5) s'annule, et nous avons $\cos A_1/2 = \sin A_1/2 = 0$. Dans le cas où $\tau_2 = \tau_4$, A_1 est forcément nul et le raisonnement précédent quant à la dérivabilité de A_1 reste valable. Et (E.14) devient :

$$\partial A_1 / \partial V_k = \delta_{k1} |\partial \mathcal{V}(\mathcal{T}) / \partial V_k| / [a_2 a_3 a_4 (1 + \tau_2 \tau_3)]. \quad (\text{E.15})$$

Le résultat $k \neq 1 \Rightarrow \partial A_1 / \partial V_k = 0$ est dû au fait que le vecteur $\partial \mathcal{V}(\mathcal{T}) / \partial V_k$, exprimé en (D.18), est le produit vectoriel de deux vecteurs qui sont colinéaires si $k \neq 1$.

E.4 Angles solides et courbures d'un polyèdre

Nous reprenons la relation (E.2), qui indique que $A_1(\mathbf{V})$ est la somme des angles solides A_i des tétraèdres $\mathcal{T}_i = (V_1, V_{1 \rightarrow i}, V_{1 \rightarrow i+1}, W_1)$, $i = 1, \dots, n_1$ en V_1 . Nous attirons l'attention du lecteur sur le fait que les angles solides A_i appartiennent tous à $(0, 2\pi)$, mais $A_1(\mathbf{V}) \in (0, 4\pi)$, car il mesure la surface d'une portion de sphère unité, éventuellement plus grande qu'une demi-sphère. Il n'est en pratique pas nécessaire, et serait très coûteux, de calculer directement $A_1(\mathbf{V})$ à partir des termes $t(A_i)$ en faisant n_1 appels à la fonction arctan. Nous envisageons simplement l'évaluation de $\exp(i A_1(\mathbf{V})/2)$ qui spécifie $A_1(\mathbf{V})$ sans ambiguïté :

$$\exp(i A_1(\mathbf{V})/2) = \prod_{j=1}^{n_1} \exp(i A_j/2). \quad (\text{E.16})$$

Les dérivées partielles de $A_1(\mathbf{V})$ s'obtiennent en sommant des termes de type (E.10), (E.14) ou (E.15). Le nombre de termes à sommer est de n_1 pour les dérivées partielles par rapport à V_1 , et de 2 pour les autres, soit le nombre d'apparitions de V_1 et de chaque voisin $V_{1 \rightarrow i}$ dans la décomposition (E.2). Enfin, les dérivées de $\phi(\kappa_1(\mathbf{V}))$ par rapport aux sommets V_k s'obtiennent à partir de $\partial A_1(\mathbf{V}) / \partial V_k$:

$$\partial \phi(\kappa_1(\mathbf{V})) / \partial V_k = -1/2 \phi'(\kappa_1(\mathbf{V})) \sin(A_1/2) \partial A_1 / \partial V_k. \quad (\text{E.17})$$

La procédure de calcul de $\mathcal{R}(\mathbf{V})$ et $\nabla \mathcal{R}(\mathbf{V})$ est finalement présentée dans le tableau E.1. Pour des raisons de lisibilité, elle ne concerne que le cas général où aucun tétraèdre $\mathcal{T}_i$ n'est singulier.

```

1. /* Calcul de  $\mathcal{R}(\mathbf{V})$  */
   Pour tout sommet  $l$ ,
     Calculer la direction  $\boldsymbol{\tau}_l$  par (E.1) et définir le point fictif  $W_l = V_l + \boldsymbol{\tau}_l$ .
     Pour  $j = 1, \dots, n_l$ ,
       Former le tétraèdre  $\mathcal{T}_{lj} = (V_l, V_{l \rightarrow j}, V_{l \rightarrow j+1}, W_l)$ .
       Calculer  $\mathcal{V}(\mathcal{T}_{lj})$ , puis  $\exp(i A_{lj}/2)$  en appliquant (E.5) et (E.6).
     Fin pour.
     En déduire  $\exp(i A_l/2)$  par (E.16), puis  $\kappa_l(\mathbf{V})$ .
   Fin pour.
   Calculer  $\mathcal{R}(\mathbf{V})$  par (V.10).

2. /* Calcul de  $\nabla \mathcal{R}(\mathbf{V})$  */
   Pour tout sommet  $l$ ,
     /* Calcul au préalable des termes  $\partial A_l / \partial V_k$  */
     Pour  $j = 1, \dots, n_l$ ,
       Calculer  $\boldsymbol{\xi}_1(\mathcal{T}_{lj})$  par (E.8),  $\boldsymbol{\xi}_2(\mathcal{T}_{lj})$  et  $\boldsymbol{\xi}_3(\mathcal{T}_{lj})$  par (E.7).
       Calculer les dérivées de  $\mathcal{V}(\mathcal{T}_{lj})$  par rapport à  $V_l, V_{l \rightarrow j}$  et  $V_{l \rightarrow j+1}$  par (D.18).
       En déduire les dérivées de  $A_{lj}$  par rapport à  $V_l, V_{l \rightarrow j}$  et  $V_{l \rightarrow j+1}$  par (E.10).
       Faire  $\partial A_l / \partial \cdot \ += \partial A_{lj} / \partial \cdot$  pour « $\cdot$ » =  $V_l, V_{l \rightarrow j}, V_{l \rightarrow j+1}$ .
       Si  $j > 1$ ,
         /* Le calcul de  $\partial A_{lj} / \partial V_{l \rightarrow j}$  est achevé */
         Calculer  $\partial \phi(\kappa_l(\mathbf{V})) / \partial V_{l \rightarrow j}$  par (E.17).
       Fin si.
     Fin pour.
     Calculer  $\partial \phi(\kappa_l(\mathbf{V})) / \partial V_{l \rightarrow 1}$  et  $\partial \phi(\kappa_l(\mathbf{V})) / \partial V_l$  par (E.17).
     /* Mise à jour de  $\nabla \mathcal{R}(\mathbf{V})$  */
     Pour  $k \in \{l\} \cup \partial l$ , faire  $\partial \mathcal{R}(\mathbf{V}) / \partial V_k \ += \partial \phi(\kappa_l(\mathbf{V})) / \partial V_k$ .
   Fin pour.

```

Tableau E.1 – Régularisation portant sur l'angle solide. Procédure de calcul de $\mathcal{R}(\mathbf{V})$ et $\nabla \mathcal{R}(\mathbf{V})$.

BIBLIOGRAPHIE

- Y. Amit et K. M. Manbeck. Deformable template models for emission tomography. *IEEE Trans. Medical Imaging*, 12 (2) : 260–268, juin 1993.
- T. Arak et D. Surgailis. Markov fields with polygonal realizations. *Prob. Theory Related Fields*, 80 : 543–579, 1989.
- J. Arvo. *Graphics Gems II*. Academic Press, 1991.
- E. Bardinet, L. D. Cohen et N. Ayache. Fitting 3-D data using superquadrics and free-form deformations. In *Proc. 12-th Int. Conf. on Pattern Recognition*, volume 1, pages 79–83, Jerusalem, Israel, octobre 1994.
- S. Basu et Y. Bresler. Tomography with unknown view angles. In *Proc. IEEE ICASSP*, volume 4, pages 2845–2848, Munich, Allemagne, avril 1997.
- X. L. Battle, G. S. Cunningham et K. M. Hanson. Tomographic reconstruction using 3D deformable models. *Phys. Med. Biol.*, 43 : 983–990, 1998.
- X. L. Battle, C. Le Rest, A. Turzo et Y. J. Bizais. 3D attenuation map reconstruction using geometrical models and free-form deformations. *IEEE Trans. Medical Imaging*, pages 1–8, mai 2000.
- M. Berger. *Geometry I and II*. Springer-Verlag, Berlin–New York, 1987.
- D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, Belmont, MA, USA, 1995.
- J. E. Besag. On the statistical analysis of dirty pictures (with discussion). *J. R. Statist. Soc. B*, 48 (3) : 259–302, 1986.
- J. E. Besag. Digital image processing : Towards Bayesian image analysis. *J. Appl. Statistics*, 16 (3) : 395–407, 1989.
- G. Bilbro, W. Snyder, S. Garnier et J. Gault. Mean-field annealing : A formalism for constructing GNC-like algorithms. *IEEE Transactions on Neural Networks*, 3 (1) : 131–138, janvier 1992.
- A. Blake et A. Zisserman. *Visual reconstruction*. The MIT Press, Cambridge, MA, USA, 1987.
- C. A. Bouman et B. Liu. Segmentation of textured images using a multiple resolution approach. In *Proc. IEEE ICASSP*, pages 1124–1127, New York, NY, USA, avril 1988.
- C. A. Bouman et K. D. Sauer. A generalized Gaussian image model for edge-preserving MAP estimation. *IEEE Trans. Image Processing*, 2 (3) : 296–310, juillet 1993.
- C. A. Bouman et M. Shapiro. A multiscale random field model for Bayesian image segmentation. *IEEE Trans. Image Processing*, 3 (2) : 162–177, mars 1994.
- Y. Bresler et A. Macovski. Three-dimensional reconstruction from projections with incomplete and noisy data by object estimation. *IEEE Trans. Acoust. Speech, Signal Processing*, ASSP-35 (8) : 1139–1152, août 1987.
- A. J. Bulpitt et N. D. Efford. An efficient 3D deformable model with a self-optimising mesh. *Image and Vision Computing*, 14 : 573–580, 1996.
- A. Buvaneshwari et P. S. Naidu. Estimation of shape of binary polygonal object from scattered field. *IEEE Trans. Image Processing*, 7 (2) : 253–257, février 1998.
- B. Chalmond. Automatic fitting of custom femoral stem prostheses : Ten years of research and clinical experience. In *Proc. 14th IEEE Conf. Engineering in Medicine and Biology Society*, pages 2056–66, Paris, 1992.
- B. Chalmond. *Éléments de modélisation pour l'analyse d'images*, volume 33 de *Mathématiques & Applications*. Springer-Verlag, Paris, 2000.
- B. Chalmond, F. Coldefy et B. Lavayssière. 3D curve reconstruction from noisy projections. In P.-J. Laurent, A. Le Méhaut et L. L. Schumaker, éditeurs, *Wavelets, Images and Surface Fitting*, pages 113–119. Peters, A. K., 1994.

- B. Chalmond, F. Coldefy et B. Lavayssière. Tomographic reconstruction from non-calibrated noisy projections in non-destructive evaluation. *Inverse Problems*, 15 : 399–411, 1999.
- P. Cinquin. *Splines unidimensionnelles sous tension et bidimensionnelles paramétrées : deux applications médicales*. Thèse de doctorat, Université de Saint Étienne, octobre 1981.
- F. Coutand, L. Garnerio et J. Fonroget. Anatomical data fusion for quantitative reconstruction in myocardial tomoscintigraphy using a spline model of the thorax organs. *IEEE Trans. Biomed. Eng.*, 44 (7) : 575–584, juillet 1997.
- G. S. Cunningham, I. Koyfman et K. M. Hanson. Improved convergence of gradient-based reconstructions using multi-scale models. *Medical Imaging : Image Processing, Proc. SPIE*, 2710 : 145–155, 1996.
- A. J. Davies. *The Finite Element Method : A First Approach*. Oxford Applied Mathematics and Computing Science Series, 1980.
- H. Delingette. General object reconstruction based on simplex meshes. Rapport de recherche 3111, INRIA, Sophia-Antipolis, février 1997.
- J.-M. Dinten. *Tomographie à partir d'un nombre limité de projections : régularisation par champs markoviens*. Thèse de doctorat, Université de Paris-Sud, janvier 1990.
- F. Eriksson. On the measure of solid angles. *Mathematics Magazine*, 63 : 184–187, 1990.
- L. A. Feldkamp, L. C. Davis et J. W. Kress. Practical cone-beam algorithm. *Journal of the Optical Society of America (A)*, 1 (6) : 612–619, 1984.
- J. A. Fessler et A. Macovski. Object-based 3-D reconstruction of arterial trees from magnetic resonance angiograms. *IEEE Trans. Medical Imaging*, 10 (1) : 25–39, mars 1991.
- S. Gautier. *Fusion de données gammagraphiques et ultrasonores. Application au contrôle non destructif*. Thèse de doctorat, Université de Paris-Sud, Orsay, décembre 1996.
- S. Geman et D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-6 (6) : 721–741, novembre 1984.
- S. Geman et D. McClure. Statistical methods for tomographic image reconstruction. In *Proceedings of the 46th Session of the ICI, Bulletin of the ICI*, volume 52, pages 5–21, 1987.
- P. Grangeat. *Analyse d'un système d'imagerie 3D par reconstruction à partir de radiographies X en géométrie conique*. Thèse de doctorat, École Nationale Supérieure des Télécommunications, Paris, 1987.
- K. M. Hanson, R. L. Bilisoly et G. S. Cunningham. Kinky tomographic reconstruction. *Medical Imaging : Image Processing, Proc. SPIE*, 2710 : 156–166, 1996.
- K. M. Hanson et G. S. Cunningham. Exploring the reliability of Bayesian reconstructions. *Medical Imaging : Image Processing, Proc. SPIE*, 2434 : 416–423, 1995.
- K. M. Hanson, G. S. Cunningham, G. R. J. Jennings et D. R. Wolf. Tomographic reconstruction based on flexible geometric models. In *Proc. IEEE ICIP*, volume 2, pages 145–147, Austin, TX, USA, novembre 1994.
- K. M. Hanson, G. S. Cunningham et R. J. McKee. Uncertainty assessment for reconstructions based on deformable geometry. *Int. J. Imaging Systems and Technology*, 8 : 506–512, 1997.
- G. T. Herman. *Image reconstruction from projections. The fundamentals of computerized tomography*. Academic Press, New York, NY, USA, 1980.
- P. J. Huber. *Robust Statistics*. Wiley, New York, NY, USA, 1981.
- A. C. Kak et M. Slaney. *Principles of Computerized Tomographic Imaging*. IEEE Press, New York, NY, USA, 1987.
- W. C. Karl, G. C. Verghese et A. S. Willsky. Reconstructing ellipsoids from projections. *CVGIP : Graphical Models and Image Processing*, 56 (2) : 124–139, 1994.
- M. Kass, A. P. Witkin et D. Terzopoulos. Snakes : Active contour models. *Int. J. Computer Vision*, 1 (4) : 321–331, 1988.
- C. Klifa. *Reconstruction tridimensionnelle d'objets. Application à la radiographie industrielle à partir d'un nombre très limité de projections*. Thèse de doctorat, École Nationale Supérieure des Télécommunications, Paris, 1991.
- A. Kuba et G. T. Herman. Discrete tomography : A historical overview. In G. T. Herman et A. Kuba, éditeurs, *Discrete Tomography : Foundations, Algorithms and Applications*, chapitre 1, pages 3–34. Birkhauser, Boston, MA, USA, 1999.

- J.-O. Lachaud et A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine three-dimensional surface extraction. *Medical Image Analysis*, 3 (2) : 187–207, 1998.
- J.-M. Lagrange. *Reconstruction tomographique à partir d'un petit nombre de vues*. Thèse de doctorat, École Normale Supérieure de Cachan, Cachan, décembre 1998.
- R. Lédée, S. Treuillet et D. Rose. Modèle harmonique de surface déformable périodiquement. Application à l'échocardiographie. In *Actes du 16^e colloque GRETSI*, pages 435–438, Grenoble, septembre 1997.
- G. Lefaix, X. Riot, P. Haigron, R. Collorec et A. Ramee. 3D modeling and deformation analysis of the vertebra with spherical harmonics. In *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 1, pages 422–425, Chicago, IL, USA, novembre 1997.
- A. Litman, D. Lesselier et F. Santosa. A level-set approach for eddy current imaging of defects in a conductive half-space. In G. Chavent et P. C. Sabatier, éditeurs, *Inverse Problems of Wave Propagation and Diffraction*, Lecture Notes in Physics, pages 250–262. Springer-Verlag, Berlin, Allemagne, 1997.
- A. Liu et B. Joe. Relationship between tetrahedron shape measures. *BIT*, 34 : 268–287, 1994.
- R. Malladi, J. A. Sethian et B. C. Vemuri. Shape modeling with front propagation : A level set approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17 (2) : 158–175, février 1995.
- A. Matheny et D. B. Goldgof. The use of three and four-dimensional surface harmonics for rigid and nonrigid shape recovery and representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17 (10) : 967–981, octobre 1995.
- P. Milanfar, W. C. Karl et A. S. Willsky. Reconstructing binary polygonal objects from projections : A statistical view. *Comput. Vision and Graphics and Image Processing*, 56 (5) : 371–391, septembre 1994.
- P. Milanfar, W. C. Karl et A. S. Willsky. A moment-based variational approach to tomographic reconstruction. *IEEE Trans. Image Processing*, 5 (3) : 459–470, février 1996.
- P. Milanfar, G. C. Verghese, W. C. Karl et A. S. Willsky. Reconstructing polygons from moments with connections to array processing. *IEEE Trans. Signal Processing*, 43 (2) : 432–443, février 1995.
- B. Mirtich. Fast and accurate computation of polyhedral mass properties. *Journal of Graphic Tools*, 1 (2) : 31–50, 1996.
- A. Mohammad-Djafari. Binary polygonal shape image reconstruction from a small number of projections. *Elektrik*, 5 (1) : 127–138, 1997.
- A. Mohammad-Djafari et G. Demoment. Maximum entropy reconstruction in X ray and diffraction tomography. *IEEE Trans. Medical Imaging*, MI-7 (4) : 345–354, 1988.
- A. Mohammad-Djafari et Y. Khayi. Array processing techniques and shape reconstruction in tomography. In *Proc. IEEE ICASSP*, volume II, pages 3785–3789, Munich, Allemagne, avril 1997a.
- A. Mohammad-Djafari et Y. Khayi. Évaluation de méthodes de traitement d'antenne en reconstruction d'image appliquée au contrôle non destructif. In *Actes du 16^e colloque GRETSI*, pages 135–139, Grenoble, septembre 1997b.
- A. Mohammad-Djafari et C. Soussen. Compact object reconstruction. In G. T. Herman et A. Kuba, éditeurs, *Discrete Tomography : Foundations, Algorithms and Applications*, chapitre 14, pages 317–342. Birkhauser, Boston, USA, 1999.
- F. Natterer. *The mathematics of computerized tomography*. Wiley, 1986.
- J. A. Nelder et R. Mead. A simplex method for function minimization. *Computer Journal*, 7 : 308–313, 1965.
- M. Nikolova. Markovian reconstruction using a GNC approach. *IEEE Trans. Image Processing*, 8 (9) : 1204–1220, septembre 1999.
- M. Nikolova et A. Mohammad-Djafari. Eddy current tomography using a binary Markov model. *Signal Processing*, 49 : 119–132, 1996.
- S. Osher et J. A. Sethian. Fronts propagating with curvature-dependent speed : Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79 : 12–49, 1988.
- C. Pellot, A. Herment, M. Sigelle, P. Horain, H. Maître et P. Peronneau. A 3D reconstruction of vascular structures from two X-ray angiograms using an adapted simulated annealing algorithm. *IEEE Trans. Medical Imaging*, 13 (1) : 48–60, mars 1994.
- P. Pérez et F. Heitz. Restriction of a Markov random field on a graph and multiresolution statistical image modeling. *IEEE Trans. Inf. Theory*, 42 : 180–190, janvier 1996.
- J. L. Prince et A. S. Willsky. Convex set reconstruction using prior shape information. *Comput. Vision and Graphics and Image Processing*, 53 (5) : 413–427, septembre 1991.

- M. C. Rivara. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal of Numerical Methods in Engineering*, 20 : 745–756, 1984.
- M. Robini. *Inversion bayésienne et algorithmes stochastiques de type recuit. Application à la restauration d'images et à la reconstruction 3-D à partir d'un nombre très limité de projections*. Thèse de doctorat, Institut National des Sciences Appliquées, Lyon, 1998.
- D. J. Rossi et A. S. Willsky. Reconstruction from projections based on detection and estimation of objects – Parts I and II : Performance analysis and robustness analysis. *IEEE Trans. Acoust. Speech, Signal Processing*, ASSP-32 (4) : 886–906, août 1984.
- F. Santosa. A level-set approach for inverse problems involving obstacles. *Control, Optimisation and Calculus of Variations*, 1 : 17–33, janvier 1996.
- K. D. Sauer et C. A. Bouman. Bayesian estimation of transmission tomograms using segmentation based optimization. *IEEE Trans. Nuclear Sciences*, 39 : 1144–1152, février 1992.
- K. D. Sauer, J. J. Sachs et C. Klifa. Bayesian estimation of 3-D objects from few radiographs. *IEEE Trans. Nuclear Sciences*, 41 (5) : 1780–1790, octobre 1994.
- W. J. Schroeder, J. A. Zarge et W. E. Lorensen. Decimation of triangle meshes. *Computer Graphics*, 26 (2) : 65–70, juillet 1992.
- M. Senasli, L. Garnerio, A. Herment et E. Mousseaux. 3D reconstruction of vessel lumen from very few angiograms by dynamic contours using a stochastic approach. *Graphical Models and Image Processing*, 62 : 105–127, 2000.
- W. Snyder, Y.-S. Han, G. Bilbro, R. T. Whitaker et S. Pizer. Image relaxation : Restoration and feature extraction. *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-17 (6) : 620–624, juin 1995.
- F. Solina et R. K. Bajcsy. Recovery of parametric models from range images : The case for superquadrics with global deformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-12 (2) : 131–147, février 1990.
- M. Soumekh. Reconstruction from incomplete data in echo diffraction imaging. In *Proc. IEEE ICASSP*, pages 1304–1307, New York, NY, USA, avril 1988.
- C. Soussen et A. Mohammad-Djafari. Approche multirésolution pour la reconstruction 3D de défaut en tomographie X. In *Actes du 17^e colloque GRETSI*, pages 1197–1200, Vannes, septembre 1999a.
- C. Soussen et A. Mohammad-Djafari. Multiresolution approach to the estimation of the shape of a 3D compact object from its radiographic data. In F. Prêteux, A. Mohammad-Djafari et E. R. Dougherty, éditeurs, *Proc. SPIE, Mathematical Modeling, Bayesian Estimation, and Inverse Problems*, pages 150–160, Denver, CO, USA, juillet 1999b.
- C. Soussen et A. Mohammad-Djafari. Recovery of moments of a compact homogeneous object from tomographic projection data, soumis à *Graphical Models and Image Processing*, GPI-L2S, 1999c.
- L. H. Staib et J. S. Duncan. Deformable fourier models for surface finding in 3D images. *Visualization in Biomedical Computing*, 1808 : 90–104, 1992.
- L. H. Staib et J. S. Duncan. Model-based deformable surface finding for medical images. *IEEE Trans. Medical Imaging*, 15 (5) : 720–731, octobre 1996.
- S. Teboul, L. Blanc-Féraud, G. Aubert, M. Barlaud et J. Darcourt. Contours déformables et reconstruction tomographique en imagerie médicale. In *Actes du 16^e colloque GRETSI*, pages 1495–1498, Grenoble, septembre 1997.
- A. Tikhonov et V. Arsenin. *Méthodes de résolution de problèmes mal posés*. Éditions MIR, Moscou, Russie, 1976.
- H. G. Timmer et J. M. Stern. Computation of global geometric properties of solid objects. *Computer Aided Design*, 12 (6) : 301–304, novembre 1980.
- R. T. Whitaker. A level-set approach to 3D reconstruction from range data. *Int. J. Computer Vision*, 29 (3) : 203–231, 1998.

Nom : SOUSSEN

Prénom : Charles

Titre : Reconstruction 3D d'un objet compact en tomographie.

Title : Tomographic reconstruction of a 3D compact object.

Résumé :

Le contexte de ce travail est la reconstruction 3D d'images binaires en tomographie X à partir d'un faible nombre de vues. En contrôle non destructif, une première expertise peut révéler la présence d'une seule zone défectueuse, ou occlusion dans un milieu homogène. Le défaut est alors considéré comme un objet compact et homogène. Sa reconstruction est sous-déterminée car les projections sont limitées en nombre et en angles. Il est alors indispensable d'apporter de l'information *a priori* sur l'objet et/ou le milieu pour obtenir une solution acceptable. Une approche possible est la modélisation probabiliste de cette information et la méthodologie bayésienne. Nous distinguons deux classes de méthodes. Les premières, plus classiques, modélisent l'image par un champ binaire et estiment les valeurs de ses voxels à partir des données. Les secondes modélisent directement le *contour* de l'objet, et réalisent son estimation à partir des données. Compte tenu du contexte, nous optons pour ces dernières car elles permettent de limiter le nombre de paramètres et d'exploiter explicitement la compacité de l'objet. La régularisation porte alors sur la douceur locale du contour. Après une présentation des deux approches, de leurs avantages et limitations, nous étudions les approches par contour en sélectionnant des paramétrisations adaptées. Nous optons pour des modélisations locales, dont chaque paramètre ne contrôle qu'une portion du contour, afin de reconstruire des surfaces complexes. En particulier, nous utilisons la représentation *polyédrique*. Ce choix est conforté par des aspects algorithmiques, et notamment l'étude des projections du polyèdre. Nous concevons une méthode originale de reconstruction qui s'appuie sur l'estimation directe de la position des sommets du polyèdre à partir des données, sans recourir à une représentation par voxel de la scène examinée. Nous discutons son efficacité et sa capacité à reconstruire des objets de forme complexe de façon rapide.

Abstract :

We consider the 3D reconstruction of binary images from a few X-ray tomographic data. In nondestructive testing, a first evaluation may determine the presence of a *single* fault in an homogeneous material. The fault is then considered as a compact homogeneous object. Its reconstruction is underdetermined due to the low number and the limited angles of the projections. One then needs to provide prior information on the region of interest to obtain an accurate solution. A possible approach to regularize the problem is to use the probabilistic modeling of this information in the Bayesian inference framework. We distinguish two classes of methods. The most classical ones model the image as a binary field and then estimate the values of its voxels from the data. The others describe and estimate the object surface from the data. The latter are the most natural in our context since they take advantage of the compacity assumption, and they significantly reduce the number of parameters. We present the advantages and limitations of both strategies, and then study the contour-based models. Selected prior information favors local smoothness of the model. To reconstruct complex surfaces, we choose local parametric models, whose parameters only control specific portions of the surface. We use spline representations, and we especially show the interest of the *polyhedral* modeling. We elaborate an original reconstruction method which directly estimates the positions of the polyhedron vertices from the data, without a voxel representation of the region of interest. Algorithmic aspects derive from the properties of the model projections. We study the performance of the method and we show its ability to reconstruct complex shapes in a limited time of computation.

Mots clés : tomographie à rayons X, reconstruction 3D, surface fermée, modèle polyédrique, surface déformable, régularisation.

Keywords : X-ray tomography, 3D reconstruction, closed surface, polyhedral modeling, deformable template, regularization.