Discrete curvature: theory and applications

Laurent Najman, Pascal Romon

Courbure discrète : théorie et applications

Rencontre organisée par :
Laurent Najman et Pascal Romon

18-22 novembre 2013

Table des matières

Introduction

Laurent Najman and Pascal Romon

The present volume contains the proceedings of the 2013 Meeting on discrete curvature, held at CIRM, Luminy, France. The aim of this meeting was to bring together researchers from various backgrounds, ranging from mathematics to computer science, with a focus on both theory and applications. With 27 invited talks and 8 posters, the conference attracted 70 researchers from all over the world. The challenge of finding a common ground on the topic of discrete curvature was met with success, and these proceedings are a testimony of this work.

Discrete curvature has been a fast-growing topic and common denominator in many fields in the past decade. Indeed, many applications and theoretical constructions are related to or rely on some notion of discrete curvature or one of its avatars, such as the Laplace-Beltrami operator. More interestingly, new concepts and new approaches have emerged, sometimes from seemingly disconnected fields, allowing a better understanding of curvature in the discrete realm, as well as new ways to tackle applications. In parallel, new challenges arise in computer science that require more sophisticated theoretical apparatus, often using the latest theoretical developments.

Discrete curvature may arise on discretized surfaces, raising the problem of convergence to the smooth model. In that case, the curvature—or more appropriately the curvatures—need to be defined, and are the goal as well as an obstacle to convergence (Morvan & Sun, Fu, Tai). Its definition allows countless applications (Bac *et al.*, Olsson & Boykov). Discrete curvature may also appear as geometrically relevant quantity in a discrete space otherwise disconnected from actual smooth geometry, such as a graph (Keller). What we have seen in this meeting are different definitions and concepts of discrete curvature, suited to different problems and settings. However, they all share the common trait of defining a notion according to its geometric consequences and properties.

One example of recent development is the increasing role played by optimal transportation, an old problem with new aspects which applies particularly well to the discrete setup. The Wasserstein metric may be used to define curvature and deduce combinatorial, functional and topological information (Bauer *et al.*, Maas), but also to compare shapes (Alliez *et al.*, Memoli). This comparison principle is found also on polytopes (Baird), and analogous ideas appear in deformation and shape matching using various energies (Cremers *et al.*, Sorkine). And it relates to the metric approach to curvature (Saucan).

The Laplace-Beltrami operator and energy functional are linked to the curvature and play a key role in understanding the geometry and spectrum, and of course in countless applications, such as segmentation, inpainting (Stuehmer), curvature flows (Aflalo *et al.*, Boykov). A more profound approach to this operator can be achieved through exterior differential calculus (Memari, Leok), which seeks to preserve (some of) the structural properties of the continuum. Other functionals may intervene such as the Willmore functional or the Hilbert-Einstein functional (Izmestiev) to characterize the geometry.

A similar albeit different approach to differential calculus comes from integrable theory, developed in particular by the German school. The choice of a particular mesh (circular, quad-based) implies analogous properties to those of special surfaces (e.g. minimal or constant mean curvature surfaces) or special parametrization such as conformal maps. This yields natural and

robust definitions for the curvature (Hoffmann) and applies remarkably to architectural constructions (Pottmann); at the same time, it relates to discrete complex analysis (Bobenko & Günther, Skopenkov).

Digital geometry also leads back to curvature, which can be defined on pixels and voxels in a very geometric way (Lachaud), and applies to topological problems (Kenmochi *et al.*) and flows (Imiya). One of the crucial issues there, as in many of the above cases (but not all), is the one of consistency or convergence properties, thus allowing to compute smooth quantities by refinement.

The success of this meeting is an encouragement to go further ahead. In addition to these proceedings, we plan to edit in the near future a survey of the different approaches to discrete curvature, based on what was presented here at CIRM, and reaching beyond. Such a resource will be very valuable to both researchers and students wishing to enter this rich field, and will help to disseminating and expanding the theory.

LIGM , ESIEE, 2, boulevard Blaise Pascal, Cité Descartes, BP 99, 93162 Noisy-le-Grand cedex, France • l.najman@esiee.fr

Université Paris-Est, LAMA (UMR 8050), UPEMLV, F-77454, Marne-la-Vallée, France • pascal.romon@u-pem.fr

Curvature measures, normal cycles and asymptotic cones

Xiang Sun and Jean-Marie Morvan

Abstract

The purpose of this article is to give an overview of the theory of the *normal cycle* and to show how to use it to define a *curvature measures* on singular surfaces embedded in an (oriented) Euclidean space $\mathbb{E}^3$. In particular, we will introduce the notion of *asymptotic cone* associated to a Borel subset of $\mathbb{E}^3$, generalizing the *asymptotic directions* defined at each point of a smooth surface. For simplicity, we restrict our singular subsets to polyhedra of the 3-dimensional Euclidean space $\mathbb{E}^3$. The coherence of the theory lies in a convergence theorem: If a sequence of polyhedra (P_n) tends (for a suitable topology) to a smooth surface S, then the sequence of curvature measures of (P_n) tends to the curvature measures of S. Details on the first part of these pages can be found in [6].

1. Smooth surfaces and polyhedra

1.1. **Smooth surfaces.** Let us deal with the local *Riemannian* geometry of submanifolds. We endow $\mathbb{E}^3$ with its scalar product $< .,. >$ and its associated Levi-Civita connexion $\tilde{\nabla}$. In our context, an (oriented) *smooth surface S* of the (oriented) Euclidean space $\mathbb{E}^3$ means a 2-dimensional(oriented) C^2-manifold embedded in $\mathbb{E}^3$. We will only deal with closed oriented surfaces bounding a compact domain in $\mathbb{E}^3$. Such surfaces are endowed with a Riemannian structure induced by $< .,. >$. We still denote by $< .,. >$ their metric. With such a structure, the embedding of S in $\mathbb{E}^3$ becomes an isometric embedding. We denote by TS the tangent bundle of S and by ξ be the unit normal vector field compatible with the orientation. The *Weingarten tensor $A : TS \to TS$*, is the symmetric endomorphism defined for all X in TS by $A(X) = -\tilde{\nabla}_X \xi$. The second fundamental form of S is the symmetric tensor defined for every X, Y in TS by $h(X,Y) = < A(X), Y >$. The real function $H = \frac{1}{2}$trace A is called the mean curvature of S, and the real function $G = \det A$ is called the Gauss curvature of S.

1.2. **Polyhedra.** We consider here (triangulated) polyhedra as 2-dimensional piecewise linear surfaces embedded in $\mathbb{E}^3$. These polyhedra will be closed, bounding a 3-dimensional domain. The area of the triangles, the length of the edges, the solid angle at vertices and the angle of incident triangles describe their geometry. If P is a polyhedron, we denote by **V**, resp. **E**, resp. **T** the set of its vertices, resp. edges, resp. triangles. Let us give precise definitions of angles.

Definition 1. Let P be a (triangulated) polyhedron in $\mathbb{E}^3$.
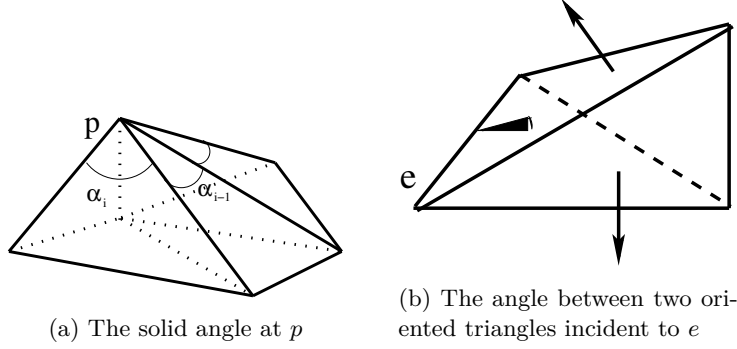
(1) The solid angle of a vertex p of P is defined by

$$\alpha_p = \sum_i \alpha_{i_p},$$

where the α_{i_p} are the angles at p of the triangles t_i incident to p.

(a) The solid angle at p

(b) The angle between two oriented triangles incident to e

(2) The angle $\angle(e)$ between two oriented triangles t_1 and t_2 (with a common edge e) is the angle angle defined by their oriented normals.

2. A GLOBAL DEFINITION OF THE MEAN CURVATURE OF A CONVEX POLYHEDRON

As an example, we discuss here a possible definition of the *global mean curvature* of a *convex* polyhedron, by using the well known *Steiner formula* of the volume of tubes (see for instance [6]). This formula gives the behavior of the volume $\mathrm{Vol}_3(\mathcal{K}_\epsilon)$ of the tube $\mathcal{K}_\epsilon$ of radius ϵ of a convex subset $\mathcal{K}$ of $\mathbb{E}^3$. By definition, $\mathcal{K}_\epsilon$ is the set of points at distance less or equal to ϵ of $\mathcal{K}$. Its volume is a polynomial in ϵ, whose coefficients (called the *Quermassintegrale* of Minkowski) depend only on $\mathcal{K}$.

Theorem 2. (1) *Let $\mathcal{K}$ be a convex subset of $\mathbb{E}^3$. Then for all $\epsilon \geq 0$, there exist constants $\Phi_k(\mathcal{K})$, $0 \leq k \leq 3$ such that*

$$(2.1) \qquad \mathrm{Vol}_3(\mathcal{K}_\epsilon) = \sum_{k=0}^{3} \Phi_k(\mathcal{K})\epsilon^k.$$

In particular, the coefficients $\Phi_k(\mathcal{K})$'s can be explicitly computed in special situations :
(2) *If $\mathcal{K}$ is a convex domain with smooth boundary $S = \partial\mathcal{K}$, then*

$$(2.2) \qquad Vol_3(\mathcal{K}_\epsilon) = Vol_3(\mathcal{K}) + A(S)\epsilon + (\int_S H da)\epsilon^2 + \frac{4}{3}\pi\epsilon^3.$$

(3) *If $\mathcal{K}$ is a convex domain with polyhedral boundary $P = \partial\mathcal{K}$ then,*

$$(2.3) \qquad Vol_3(\mathcal{K}_\epsilon) = Vol_3(\mathcal{K}) + A(P)\epsilon + (\sum_a \angle(a)l(a))\epsilon^2 + \frac{4}{3}\pi\epsilon^3.$$

Here, $A(S)$ (resp. $A(P)$) denotes the area of S (resp. P).

Let us compare (2.2) and (2.3). By analogy with the smooth case, one can define the *global mean curvature* of a convex domain $\mathcal{K}$ with polyhedral boundary P as the coefficient $\Phi_2(\mathcal{K})$, and give the following definition, with the previous notations:

Definition 3. The global mean curvature of a convex domain with polyhedral boundary (or simply, if there is no possible confusion, the global mean curvature of a convex polyhedron) of $\mathbb{E}^3$ is the real number

$$\sum_{e\in\mathbf{E}} l(e)\angle(e).$$

We remark that it can be proved that all the Φ_k satisfy the following basic properties: If A and B are convex subsets such that $A \cup B$ and $A \cap B$ are convex, then for all $k \in \{0, 1, 2, 3\}$,

$$(2.4) \qquad \Phi_k(A \cup B) = \Phi_k(A) + \Phi_k(B) - \Phi_k(A \cap B).$$

This implies that, in order to compute the curvatures of a complicated convex subset, on can decompose it into simple convex subsets and apply (2.4). Moreover, if C_n is a sequence of convex subsets tending to a convex subset C in the Hausdorff sense, then $\lim_n \Phi_k(C_n) = \Phi_k(C)$. In

particular, if P_n is a sequence of convex polyhedra tending to a convex surface S in the Hausdorff sense,

$$\lim_n \sum_{e_{k_n} \in \mathbf{E}_n} l(e_{k_n}) \angle(e_{k_n}) = \int_S H da,$$

where da denotes the area form on S.

3. Invariant differential 2-forms and normal cycle

The theory of the *normal cycle* is still in progress. It has been introduced theory by P. Wintgen and M. Zahle to give a general method to define curvatures of a large class of objects, without any assumptions of smoothness or convexity, [7], [8]. It has been successfully developed by many authors, in particular [4], [5]. Curvature measures are defined as integrals of invariant differential forms on a *generalized unit normal bundle*. The *normal cycle* of a singular object is this generalized unit normal bundle. Let us be more precise.

(1) First of all, one the following differential 2-forms on $\mathbb{E}^3 \times \mathbb{E}^3$: If $(x_1, x_2, x_3, y_1, y_2, y_3)$ are the standard coordinates on $\mathbb{E}^3 \times \mathbb{E}^3$ identified with $T\mathbb{E}^3$, we put

$$\omega_G = y_1 dy_2 \wedge dy_3 + y_2 dy_3 \wedge dy_1 + y_3 dy_1 \wedge dy_2;$$

(3.1)
$$\omega_H = y_1(dx_2 \wedge dy_3 + dy_2 \wedge dx_3) + y_2(dx_3 \wedge dy_1 + dy_3 \wedge dx_1)$$
$$+ y_3(dx_1 \wedge dy_2 + dy_1 \wedge dx_2),$$

where $\wedge$ is the exterior product of differential forms.

(2) Then, one defines the normal cycle. The theory being very general, we describe here this construction in very special cases: closed smooth surfaces bounding a domain, compact convex domains and closed polyhedra bounding a domain of $\mathbb{E}^3$.

• Let S be a (closed) surface bounding a domain D of $\mathbb{E}^3$, the unit normal bundle of D is the manifold

$$ST^\perp D = \{(p, \xi_p), p \in S, \xi_p \text{ unit normal vector at } p, \}$$

endowed with the orientation induced by the one of D. The *normal cycle* $N(D)$ of D is nothing but the 2-current canonically associated to $ST^\perp D$: If ω is any 2-differential form defined on $\mathbb{E}^3 \times \mathbb{E}^3$, the duality bracket $<,>$ is given by $< ST^\perp D, \omega >= \int_{ST^\perp D} \omega$.

• Let C be a compact convex domain of $\mathbb{E}^3$. The normal cone $\mathcal{C}_p(C)$ of a point p of C is the set of unit vectors (p, ξ_p) such that

$$\forall q \in C, \vec{pq} \cdot \xi_p \leq 0.$$

The *normal cone* $\mathcal{C}(C)$ of C is the union of the $\mathcal{C}_p(C)$, when p runs over C. The *normal cycle* $N(C)$ of C is nothing but the 2-current associated to $\mathcal{C}(C)$ endowed with its canonical orientation.

• A crucial property of the normal cycle is its the additivity: if A and B are subsets of $\mathbb{E}^3$ admitting a normal cycle, and such that $A \cap B$ admits a normal cycle, then, $N(A \cup B)$ admits a normal cycle and

(3.2)
$$N(A \cup B) = N(A) + N(B) - N(A \cap B).$$

Since a compact 3-polyhedron is the union of (convex) tetrahedra, triangles, edges and vertices, we define the normal cycle of a polyhedron by decomposing it into (convex) simplices, and we apply (3.2). Of course, the result is independent of the decomposition into convex subsets.

We will now define curvature measures in an unified way, which will, in some sense, generalize our previous definitions in the pointwise and global situations, described in the previous paragraphs. These measures will be defined as measures on $\mathbb{E}^3$. We denote by $\mathcal{B}$ the set of Borel subsets of $\mathbb{E}^3$.

Definition 4. Let M be a compact smooth surface or a polyhedron of $\mathbb{E}^3$ bounding a domain D.

(1) The Gaussian curvature measure of M is defined for every $B \in \mathcal{B}$ by

$$\Phi_M^G(B) = < N(D), \chi_{B \times \mathbb{E}^3} \omega_G > .$$

5

(2) The mean curvature measure of M is defined for every $B \in \mathcal{B}$ by

$$\Phi_M^H(B) = \; < N(D), \chi_{B \times \mathbb{E}^3} \omega_H >,$$

where $\chi_{B \times \mathbb{E}^3}$ denotes the characteristic function of $B \times \mathbb{E}^3$ in $\mathbb{E}^3 \times \mathbb{E}^3$.

Let us give the explicit expression of these curvature measures for smooth surfaces and for polyhedra, (see [5] for instance).

Theorem 5.   (1) *Let S be a smooth closed surface of $\mathbb{E}^3$.*
     (a) *The Gaussian curvature measure of S is defined for every $B \in \mathcal{B}$ by*

$$\Phi_S^G(B) = \int_{B \cap S} G da.$$

     (b) *The mean curvature measure of S is defined for every $B \in \mathcal{B}$ by*

$$\Phi_M^H(B) = \int_{B \cap S} H da.$$

(2) *Let P be a closed polyhedron of $\mathbb{E}^3$.*
     (a) *The Gaussian curvature measure of P is defined for every $B \in \mathcal{B}$ by*

$$\Phi_S^G(B) = \sum_{v \in \mathbf{V} \cap B} \alpha_v.$$

     (b) *The mean curvature measure of P is defined for every $B \in \mathcal{B}$ by*

$$\Phi_P^H(B) = \sum_{e \in \mathbf{E} \cap B} l(e \cap B) \beta(e),$$

     where $\beta(e) \in [-\pi, \pi]$ is the angle between the normals to the triangles incident on e. The sign of $\beta(e)$ is positive if e is convex and negative if e is concave.

In [5], J. Fu proved a theorem related to the convergence of these curvatures. Details on the proof and extensions can be founded in [1], [3]. We cite one of these extensions. We recall that a polyhedron is said to be *closely inscribed* in a surface S if its vertices belong to S, and if the orthogonal projection of P to S is a bijection. If B be the relative interior of an union of triangles, we denote by $pr(B)$ the orthogonal projection of B on S.

Theorem 6. *Let P be a triangulated polyhedron closely inscribed in a smooth surface S. Let B be the relative interior of an union of triangles. Then,*

$$|\Phi_P^G(B) - \Phi_S^G(pr(B))| \le C_S \mathbf{K} \epsilon; |\Phi_P^H(B) - \Phi_S^H(pr(B))| \le C_S \mathbf{K} \epsilon,$$

where C_S is a constant depending on the geometry of S, and

$$\mathbf{K} = \sum_{t \subset \overline{B}} cr(t)^2 + \sum_{t \subset \overline{B}, t \cap \partial B \neq \emptyset} cr(t), \epsilon = \max\{ cr(t), t \in \mathbf{T} \cap B\},$$

$cr(t)$ denoting the circumradius of the triangle t.

In other words, if P "approximates" S, the curvatures of P "approximate" the curvatures of S.

4. ASYMPTOTIC CONES

We denote by $\Xi(\mathbb{E}^3)$ the $\mathcal{C}^\infty$-module of smooth vector fields on $\mathbb{E}^3$, and by $\Xi_S(\mathbb{E}^3)$, (resp. $\Xi_P(\mathbb{E}^3)$) the restriction to S (resp. P) of smooth vector fields on $\mathbb{E}^3$.

Definition 7. If $Z \in \Xi(\mathbb{E}^3)$, the asymptotic 2-form associated to Z is the differential 2-form of $\mathbb{E}^3 \times \mathbb{E}^3$ defined at any point $(p, n) \in \mathbb{E}^3 \times \mathbb{E}^3$ by

(4.1) $$\mathbf{h}_{(p,n)}^Z = (n \times Z) \wedge Z,$$

where $n \times Z$ is the cross product in $\mathbb{E}^3$ of the vector field n (identified with the point n) and the vector field Z.

In this definition, we have used the usual identification of vectors and 1-forms by the standard scalar product. Remark that n is always in the kernel of $\mathbf{h}^Z_{(p,n)}$. Using the duality between 2-forms and 2-currents, we will evaluate $\mathbf{h}^Z$ on the normal cycle of Borel sets of any smooth surfaces S in $\mathbb{E}^3$. Let $\mathbf{G} : S \to \mathbb{E}^3 \times \mathbb{E}^3$ be its *Gauss map* defined for every $p \in S$ by $\mathbf{G}(p) = (p, \xi_p)$. The following is a simple computation, (see [1] or [3] for instance). We denote by pr_{TS} the orthogonal projection on the tangent space of S.

Theorem 8. (1) *Let S be a smooth surface in $\mathbb{E}^3$ bounding a domain D, and $p \in S$.*
   (a) *For every vector field $Z \in \Xi_S(\mathbb{E}^3)$,*

(4.2) $$\mathbf{G}^*_p(\mathbf{h}^Z) = h_p(pr_{TS}Z, pr_{TS}Z)da.$$

   (b) *Let B be a Borel subset of $\mathbb{E}^3$. For any $Z \in \Xi_S(\mathbb{E}^3)$,*

(4.3) $$< N(D), \chi_{B \times \mathbb{E}^3}\mathbf{h}^Z > = \int_{B \cap S} h(pr_{TS}Z, pr_{TS}Z)da.$$

(2) *Let P be a polyhedron in $\mathbb{E}^3$ bounding a domain D. For any Borel subset B in $\mathbb{E}^3$ and any constant vector field Z of $\mathbb{E}^3$,*

(4.4)
$$< N(D), \chi_{B \times \mathbb{E}^3}\mathbf{h}^Z > = \sum_{e \in \mathbf{E}} \frac{l(e \cap B)}{2}[(\angle(e) - \sin\angle(e)) < Z, e^+ >^2$$
$$+ (\angle(e) + \sin\angle(e)) < Z, e^- >^2],$$

where e^+ (resp. e^-) is the normalized sum (resp. difference) of the unit outward normal vectors to the triangles incident to e.

Let M be any smooth surface or polyhedron bounding a domain D, and let $Z \in \Xi_M(\mathbb{E}^3)$. We give the following definition:

Definition 9. The asymptotic curvature measure $\mu_{X,M}$ associated to M and the vector field Z is the signed Borel measure defined for any Borel subset $B \subset \mathbb{E}^3$ by:

(4.5) $$\mu_{Z,M}(B) = < N(D), \chi_{B \times \mathbb{E}^3}\mathbf{h}^Z > .$$

Now let us generalize the notion of asymptotic directions of smooth surfaces to integral currents. Let $D_2(\mathbb{E}^3 \times \mathbb{E}^3)$ denote the space of integral 2-currents. For every $C \in D_2(\mathbb{E}^3 \times \mathbb{E}^3)$, we define the following quadratic function ϕ on the module $\Xi(\mathbb{E}^3)$:

(4.6)
$$\phi_C : \Xi(\mathbb{E}^3) \to \mathbb{R}$$
$$X \mapsto < C, X \wedge (n \times X) > .$$

Moreover, identifying the space of constant vector fields $Z \in \Xi_{B \cap S}(\mathbb{E}^3)$ with $\mathbb{E}^3$, we can define for every $C \in D_2(\mathbb{E}^3 \times \mathbb{E}^3)$, the quadratic function ϕ^c_C as the restriction to $\mathbb{E}^3$ of ϕ_C. This leads to the following

Definition 10. The asymptotic cones associated to the current $C \in D_2(\mathbb{E}^3 \times \mathbb{E}^3)$ are the subsets defined by:

(4.7) $$\mathcal{C}_C = \{Z \in \Xi(\mathbb{E}^3), \phi_C(Z) = 0\}, \text{and } \mathcal{C}^c_C = \mathcal{C}_C \cap \mathbb{E}^3.$$

If M is a surface or a polyhedron bounding D, we put

$$\mathcal{C}_B(M) = \mathcal{C}_{N(B \cap D)} \text{ and } \mathcal{C}^c_B(M) = \mathcal{C}_B(D) \cap \mathbb{E}^3.$$

With the previous notations, using Theorem 8, we get

(4.8) $$\mathcal{C}_B(S) = \{Z \in \Xi(\mathbb{E}^3), \int_{B \cap S} h(pr_{TS}Z, pr_{TS}Z)da = 0\},$$

where h is the second fundamental form of S. Remember that classically, one says that a vector field X tangent to S is asymptotic if

(4.9) $$h(X, X) = 0.$$

It is then natural to define the asymptotic cone over each point $p \in S$ as the cone in $T_p\mathbb{E}^3$ defined by:

(4.10) $$\mathcal{C}_p(S) = \{Z \in \Xi_S(\mathbb{E}^3), h_p(pr_{T_pS}Z, pr_{T_pS}Z) = 0\},$$

where h_p denotes the second fundamental form of S at the point p. Each $\mathcal{C}_p(S)$ is a degenerated cone (the union of two planes), each one being spanned by an asymptotic direction and by the normal of the surface at p. If P is a polyhedron and B is a Borel subset of $\mathbb{E}^3$, by Theorem 8, we characterize the asymptotic cone $\mathcal{C}_B^c(P)$ of P over B as follows:

(4.11)
$$\mathcal{C}_B^c(P) = \{Z \in \mathbb{E}^3, \sum_{e \text{ edge of T}} \frac{l(e \cap B)}{2}[(\angle(e) - \sin \angle(e)) < Z, e^+ >^2$$
$$+ (\angle(e) + \sin \angle(e)) < Z, e^- >^2] = 0\}.$$

Another simpler cone associated to B demanding less computation can be given by the equation:

(4.12)
$$\sum_{e \in \mathbf{E}} l(e \cap B)\angle(e) < Z, e >^2 = 0.$$

Both of them can be used in different contexts. In the smooth case the second one (4.12) is obtained by replacing h by $h \circ j$ in (4.9) where j denotes a rotation of $\frac{\pi}{2}$ in the tangent plane.

With the same techniques as in the proof of Theorem 6, see [5], [1], [3], [2], we get the following result. In the assumptions of the theorem, we introduce the notion of *fatness* of a triangulation. If P is a triangulated polyhedron, the *fatness* of each of its triangle t is the real number $\frac{A(t)}{l^2}$ where $A(t)$ denotes the area of t and l the length of its longest edge. The *fatness* of P is the infimum of the fatness of its triangles. Roughly speaking, the fatness of P is not too small if the angles of its triangles are not too small...

Theorem 11. *Let S be a smooth surface of $\mathbb{E}^3$, let (P_n) be a sequence of (triangulated) polyhedra closely inscribed in S, and let Z be a constant vector field. Suppose that the Hausdorff limit of P_n is S when n tends to infinity, and the fatness of P_n is uniformly bounded from below by a strictly positive constant. Then, for any $Z \in \mathbb{E}^3$,*

(4.13)
$$\lim_{n \to \infty} \mu_{Z,P_n} = \mu_{Z,S},$$

for the weak convergence of measures.

5. Examples

Let us consider the following hyperbolic surface S given by equation:

(5.1)
$$z = 1.1x^2 - y^2.$$

Let us draw a triangulation T on it. First of all we select 4 points $p_1, ..., p_4$ on S and build the points $q_1, ..., q_4$ on T such that $pr\ q_i = p_i$, $i = 1, 2, 3, 4$. Then we draw 4 balls $B_1, ..., B_4$ (with the same radius) centered at $q_1, ..., q_4$ respectively. These balls are the Borel sets from which we deduce 4 cones $C_1, ..., C_4$ computed by formula 4.11. These cones are centered at $q_1, ..., q_4$. We get figure 5.1.

Now we only select one point p on S, we build the point q on T such that $pr\ q = p$ and we draw the plane P spanned by the triangle containing q. Associated to some Borel subset, we build the cone C, centered at q. The intersection of P and C is reduced to 2 lines which approximated the asymptotic directions of S at p. We get figure 5.2.

Finally in figure 5.3, we plot some asymptotic lines of S and the approximated asymptotic directions computed by the previous process.
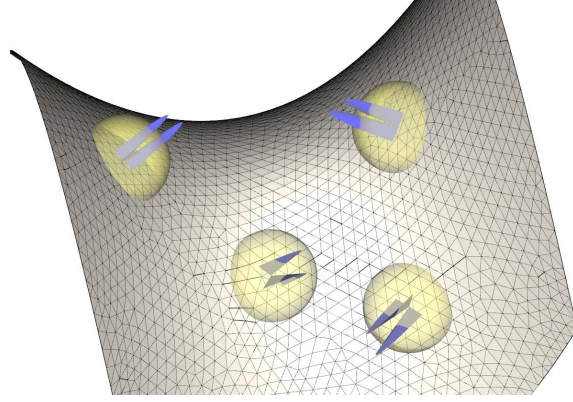
Figure 5.1. Four cones in blue in the center of their corresponding Borel sets in yellow, computed by using the triangulation.
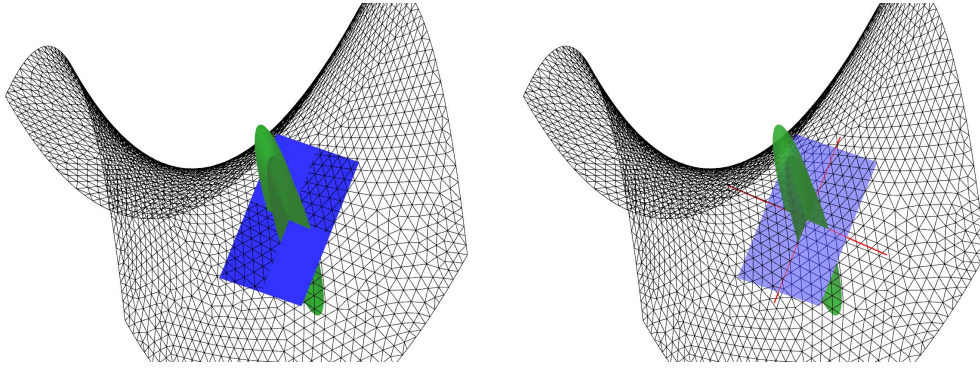
(a) The blue plane is the plane spanned by a triangle of T. It approximated the tangent plane of the smooth surface.

(b) The red lines are the intersections of the cone with the blue plane. They approximate the asymptotic directions of the surface.

Figure 5.2
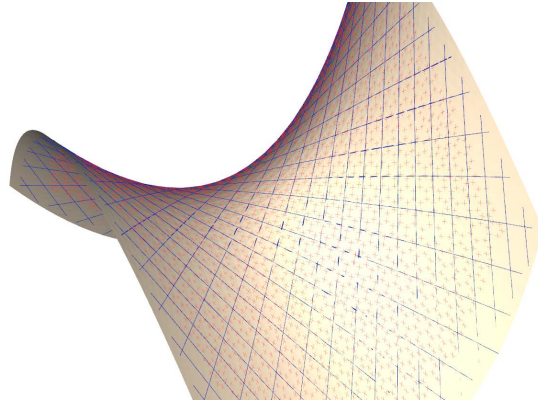
Figure 5.3. The blue lines are some asymtotic lines of S and the red lines are the approximation of asymptotic directions of S by the process described in 5.2.

REFERENCES

[1] David Cohen-Steiner and Jean-Marie Morvan. Restricted delaunay triangulations and normal cycle. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 312–321. ACM, 2003.

[2] David Cohen-Steiner and Jean-Marie Morvan. *Effective computational geometry for curves and surfaces*, chapter 4 Differential Geometry on Discrete Surfaces. Springer, 2006.

[3] David Cohen-Steiner and Jean-Marie Morvan. Second fundamental measure of geometric sets and local approximation of curvatures. *Journal of Differential Geometry*, 74(3):363–394, 2006.

[4] Joseph HG Fu. Monge-Ampère functions 1. *Indiana Univ. Math. J.*, 38:745–771, 1989.

[5] Joseph HG Fu. Convergence of curvatures in secant approximations. *Journal of Differential Geometry*, 37(1):177–190, 1993.

[6] Jean-Marie Morvan. *Generalized curvatures*, volume 2. Springer, 2008.

[7] Peter Wintgen. Normal cycle and integral curvature for polyhedra in riemannian manifolds. *Differential Geometry. North-Holland Publishing Co., Amsterdam-New York*, 1982.

[8] Martina Zähle. Integral and current representation of federer's curvature measures. *Archiv der Mathematik*, 46(6):557–567, 1986.

Visual Computing Center, King Abdullah University of Science and Technology, Saudi Arabia • xiang.sun@kaust.edu.sa

University Claude Bernard Lyon, P1, France, C.N.R.S. U.M.R. 5028, Visual Computing Center, King Abdullah University of Science and Technology, Saudi Arabia • morvan@math.univ-lyon1.fr

Piecewise linear approximation of smooth functions of two variables

Joseph H.G. Fu

Abstract

The normal cycle of a singular subset X of a smooth manifold is a basic tool for understanding and computing the curvature of X. If X is replaced by a singular function on $\mathbb{R}^n$ then there is a natural companion notion called the *gradient cycle* of f, which has been introduced by the author and by R. Jerrard. We discuss a few fundamental facts and open problems about functions f that admit gradient cycles, with particular attention to the first nontrivial dimension $n = 2$.

1. Introduction

The Federer-Fleming theory of integral currents (developed in detail in Chapter 4 of [6]) is a mathematical tool designed to extend certain notions of differential geometry to spaces with singularities. Typically it is used to study first order problems in the calculus of variations such as the Plateau problem. However, it also works spectacularly well in the study of curvature for subspaces with singularities, providing the natural setting for Federer's theory of curvature measures and its extensions [5, 16, 9]. The key idea here is that of the *normal cycle* $N(X)$ of a singular subspace X embedded in a smooth manifold M. The normal cycle is an integral current living in the tangent sphere bundle of M that functions as a substitute for the manifold of unit normals of a smooth submanifold. It has been applied effectively in surface modeling, particularly in the problem of approximating a given surface, given either formally as a smooth submanifold or empirically in terms of collections of data points, by a polyhedron [4].

Despite its many advantages, the natural scope of the theory remains murky in the sense that a clear geometric characterization of the class of sets X admitting a normal cycle is unknown. This general problem is essentially analytic. In order to study it without getting distracted by secondary topological questions, it is convenient to consider a closely related problem in which the singular subset X is replaced by a singular *function* $f : \mathbb{R}^n \to \mathbb{R}$. In this case the normal cycle $N(X)$ is replaced by the *gradient cycle* $\mathbb{D}(f)$, an integral current living in the cotangent bundle $T^*\mathbb{R}^n \simeq \mathbb{R}^n \times \mathbb{R}^{n*}$ that serves a substitute for the graph of the gradient of f (or, more correctly, its differential, although we will conflate the two in the present note). If $\mathbb{D}(f)$ exists then f is said to be a *Monge-Ampère function*. This class has been studied by the present author and his collaborator Ryan Scott [8, 10, 11] as well as by R. Jerrard [12, 13]. We describe here some basic issues and progress in the subject, with particular attention to the case $n = 2$.

2. The normal cycle and the gradient cycle

For simplicity let us take the ambient smooth manifold M to be $\mathbb{R}^n$, and assume that $X \subset \mathbb{R}^n$ is compact. The normal cycle $N(X)$ is an integral current of dimension $n - 1$ living in $\mathbb{R}^n \times S^{n-1}$

satisfying a few inevitable conditions. Let $x_1, \ldots, x_n$ be standard coordinates for $\mathbb{R}^n$ and $y_1, \ldots, y_n$ the companion coordinates for the $\mathbb{R}^{n*} \simeq \mathbb{R}^n$ that contains the sphere S^{n-1}. Then

- the support of $N(X)$ must be compact.
- $N(X)$ has boundary zero in the sense of Stokes' theorem, i.e. evaluation of $N(X)$ against any exact $(n-1)$-form vanishes.
- $N(X)$ is *Legendrian*, i.e. evaluation of $N(X)$ against any $(n-1)$-form expressible as a wedge product with the canonical 1-form $\alpha = \sum y_i \, dx_i$ vanishes.
- Finally, $N(X)$ yields the expected Morse theory of height functions restricted to X.

The precise form of the last condition is somewhat awkward, so we refrain from stating it here. The upshot is that these four conditions are enough to determine $N(X)$ uniquely. For truly pathological subsets X this current will not exist at all. It only exists for certain "tame" subsets X, but when it does exist it is defined unambiguously.

2.1. **Monge-Ampère functions.**

The companion theory for singular functions may be described in analogous terms, with the advantage that the last condition is easier to understand. Note that the geometric and the functional settings are closely related: if f is smooth and $X := \{(x, t) : x \in \mathbb{R}^n, \ t \le f(x)\}$ then $N(X)$ is the image of $\mathrm{graph}(\nabla f) \subset \mathbb{R}^n \times \mathbb{R}^n$ under the map

$$(x; \xi) \mapsto \left(x, f(x); \frac{(-\xi; 1)}{\sqrt{1 + \xi^2}} \right)$$

and conversely. Another major conjecture states that this remains true also for singular f.

In order to state the fundamental uniqueness theorem we recall that $\mathbb{R}^n \times \mathbb{R}^n$ carries a natural *symplectic 2-form*

$$\omega := \sum_{i=1}^n dx_i \wedge dy_i.$$

Theorem 1. [8, 12] *Suppose $f : \mathbb{R}^n \to \mathbb{R}$ with $\nabla f \in L^1_{loc}$. Then there exists at most one closed integral current T of dimension n in $\mathbb{R}^n \times \mathbb{R}^n$ such that*

$$(2.1) \qquad \int_T \omega \wedge \psi = 0 \qquad\qquad \text{for all } \psi \in \Omega^{n-2}(\mathbb{R}^n \times \mathbb{R}^n)$$

$$(2.2) \qquad \mathrm{volume}(T \cap \pi^{-1} K) < \infty \qquad\qquad \text{for all compact } K \subset \mathbb{R}^n$$

$$(2.3) \qquad T \cap \pi^{-1}(p) = \{(p, \nabla f(p))\} \qquad\qquad \text{for a.e. } p \in \mathbb{R}^n.$$

Here $\pi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ is the projection to the first factor. If it exists, the current T of Theorem 1 is the *gradient cycle* of f, denoted $\mathbb{D}(f)$, and f is said to be a **Monge-Ampère (MA) function**. We denote this class by $\mathrm{MA} = \mathrm{MA}(\mathbb{R}^n)$.

Condition (2.1) says that T is *Lagrangian*. The point is that if $V \subset \mathbb{R}^n \times \mathbb{R}^n$ is a smooth oriented submanifold of dimension n then V is Lagrangian in the usual sense iff the current T defined by integration over V satisfies (2.1).

The models of MA functions are the C^2 functions, with $\mathbb{D}(f) = \mathrm{graph}(\nabla f)$ (here and elsewhere we identify the manifold $\mathrm{graph}(\nabla f)$ with the current obtained by integration over it with respect to the orientation induced by the standard orientation of $\mathbb{R}^n$). In this case

$$\int_{\mathbb{D}(f)} \phi(x, y) \, dy_1 \wedge \cdots \wedge dy_n = \int_{\mathbb{R}^n} \phi(x, \nabla f(x)) \det D^2 f(x) \, dx.$$

for $\phi \in C_c(\mathbb{R}^n \times \mathbb{R}^n)$, thus motivating the name of the class. The Lagrangian condition is equivalent in this case to the calculus rule $\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_j \partial x_i}$. The area formula yields the approximate relation

$$(2.4) \qquad \mathrm{volume}(\mathbb{D}(f)) \simeq \sum_{\#I = \#J} \int \left| \det \left(\frac{\partial^2 f}{\partial x_i \partial x_j} \right)_{i \in I, j \in J} \right|.$$

In the case $n = 1$ this class is nothing new: $f \in \mathrm{MA}(\mathbb{R})$ iff the derivative f' has locally bounded variation. Alternatively, this class may be described as the set of all functions that may be expressed as $f = g - h$ where g, h are convex. Returning to the analogy with the geometry of singular subsets of $\mathbb{R}^n$, this corresponds to the fact that a curve in $\mathbb{R}^n$ has finite total curvature iff its unit tangent vector has bounded variation.

On the other hand, for $n \geq 2$ the class MA does not fit neatly into any known analytic category. For one thing, we know that $MA(\mathbb{R}^n)$ is not closed under addition for $n \geq 2$ (cf. [10]). While any $f \in MA(\mathbb{R}^2)$ must be continuous, this is not known for $n \geq 3$. While it follows directly from the definitions that any $f \in MA(\mathbb{R}^n)$ must have gradient $\nabla f \in BV_{loc}$ (the class of such functions is sometimes denoted BV_{loc}^2), it is easy to construct examples of $f \in BV_{loc}^2(\mathbb{R}^n), n \geq 2$, that are not MA. While D. Pokorný and J. Rataj [14] have recently shown that any function on $\mathbb{R}^n$ that is expressible as the difference of two convex functions must be MA, examples [10] show that not every MA function of two or more variables is of this type. Other known subclasses of MA include the Sobolev class $W_{loc}^{2,n}(\mathbb{R}^n)$ of functions with two derivatives that are locally nth power summable, and the class of all locally Lipschitz subanalytic functions.

2.2. **Strong C^2 and PL approximations.** As a consequence of Theorem 1 and the Federer-Fleming compactness theorem for integral currents, if $f_1, f_2, \cdots \in C^2(\mathbb{R}^n)$ converge in L_{loc}^1 to f_0, with $\text{volume}(\mathbb{D}(f_i) \cap \pi^{-1}K) \leq C_K$, $i = 1, 2, \ldots$ for all compact $K \subset \mathbb{R}^n$, then $f_0 \in MA$ and $\mathbb{D}(f) = \lim \mathbb{D}(f_i)$. Such f_0 is called C^2 *strongly approximable*. **All known examples of MA functions arise in this way.** Thus another fundamental conjecture states:

(2.5) $$f \in MA \implies f \text{ is } C^2 \text{ strongly approximable.}$$

Since piecewise linear (PL) functions are locally Lipschitz and subanalytic— in fact semialgebraic— these are always MA (in this case the C^2 strong approximability of any $p \in PL$ is easy to prove using the Tarski-Seidenberg theorem). On the other hand it is also easy to construct $\mathbb{D}(p)$ directly in this case [13]. For $n = 2$ this process goes as follows. Let $\mathcal{T}$ be a triangulation of $\mathbb{R}^2$ with triangles τ_i, edges σ_j and vertices ρ_k, such that p is affine on each of these elements. We construct $\mathbb{D}(p)$ as $D_2 + D_1 + D_0$, where D_i is supported over the i-skeleton of $\mathcal{T}$.

(1) Put $D_2 := \sum_i \tau_i \times \{\nabla(p|_{\tau_i})\}$. This current is Lagrangian, and satisfies (2.3), but has nonzero boundary supported above the edges σ_j.

(2) For each edge σ_j with adjacent faces τ_0, τ_1, let s_j be the line segment in $\mathbb{R}^2$ joining $\nabla(p|_{\tau_0}), \nabla(p|_{\tau_1})$. Put $D_1 := \sum_j \sigma_j \times s_j$. Since the affine functions $p|_{\tau_0}, p|_{\tau_1}$ agree along σ_j, we see that $\sigma_j \perp s_j$, which implies that D_1 is Lagrangian. Clearly $\partial D_1 = \sum_j \partial \sigma_j \times s_j - \sigma_j \times \partial s_j$; the latter terms cancel ∂D_2.

(3) It remains to cancel the former terms. For each vertex ρ_k, let $P_k \subset \mathbb{R}^2$ be the bounded polygonal region with multiplicities whose boundary is equal to the union of the oriented segments s_j corresponding to edges σ_j incident to ρ_k. Put $D_0 := \sum_k \rho_k \times P_k$, whose boundary provides the desired cancellation. Note that the addition of $D_1 + D_0$ leaves (2.3) unchanged.

We may think of the mass of D_0 (resp. D_1) as the integral of the absolute value of the Hessian of p (resp. the integral of the norm of the Hessian of p), which are in turn closely analogous to the total absolute Gauss curvature (resp. the integral of the norm of the second fundamental form).

Thus it would also be natural to take PL, instead of C^2 as the models for MA functions, and to say that f_0 is *PL strongly approximable* if the condition above holds with the C^2 functions f_i replaced by PL functions p_i. Again we conjecture

(2.6) $$f \in MA \implies f \text{ is PL strongly approximable.}$$

It is difficult (at least for us) to imagine that conjectures (2.5) and (2.6) could possibly fail, but a proof seems far away (aside from the trivial case $n = 1$). Finding ourselves in this position we must ask: are the two conjectures are equivalent? Even this problem seems difficult, although it is true for $n = 2$. This is a consequence of the following two facts.

Theorem 2 (Brehm-Kühnel [1]). *There is a universal constant C with the following property. Given $p \in PL(\mathbb{R}^2)$ there exists a sequence $C^2(\mathbb{R}^2) \ni f_1, f_2, \cdots \to p$ locally uniformly, with*

$$\limsup \text{area}(\mathbb{D}(f_i|_U)) \leq C \text{ area}(\mathbb{D}(p|_U))$$

for any relatively compact open set $U \subset \mathbb{R}^2$.

Brehm and Kühnel state this result in different language, but this is an essentially equivalent formulation. Clearly this theorem yields: PL strongly approximable $\implies$ C^2 strongly approximable.

Theorem 3 (Fu-Scott [11])**.** *Given $f \in C^2(\mathbb{R}^2)$ there exists a sequence $\mathrm{PL}(\mathbb{R}^2) \ni p_1, p_2, \cdots \to f$ locally uniformly, with*

$$\limsup \mathrm{area}(\mathbb{D}(p_i|_U)) \le \int_U 1 + 2\sqrt{2}\,\|D^2f\| + |\det D^2f|$$

for any relatively compact open set $U \subset \mathbb{R}^2$.

Recall that by (2.4)

$$\mathrm{area}(\mathbb{D}(f|_U)) \simeq \int_U 1 + \|D^2f\| + |\det D^2f|.$$

Thus Theorem 3 yields: C^2 strongly approximable $\implies$ PL strongly approximable.

2.3. **Sketch of the proof of Theorem 3.** The basic strategy is to pick an appropriate sequence of fat triangulations $\mathcal{T}_i$ of the domain of f with mesh size $\to 0$. For each i we set the values of the PL function p_i at the vertices of $\mathcal{T}_i$ equal to those of f, then extend to each triangle by linear interpolation.

The trick lies in giving meaning to the word "appropriate". If we simply take a sequence of triangulations $\mathcal{T}_i$ of the plane with mesh size $\to 0$ and uniformly positive fatness, and let p_i be the PL function obtained by linear interpolation from the values of f at the vertices of $\mathcal{T}_i$, then

$$(2.7) \qquad \limsup \mathrm{area}(\mathbb{D}(p_i|_U)) \simeq \int_U 1 + \|D^2f\| + \|D^2f\|^2,$$

the last term replacing the desired term $|\det D^2f|$. Although the first two terms are acceptable, in general the last term is too big, as may be seen in the following simple example. Let $f(x,y) = (x-y)^2$ and $\mathcal{T}_i$ be a subdivided square grid aligned with the coordinate axes. Construct the PL function p_i as we have just described, and consider the gradient cycle $\mathbb{D}(p_i) = D_2 + D_1 + D_0$ given by the procedure above. The polygons P_k making up the summand D_0 are all congruent copies of the figure
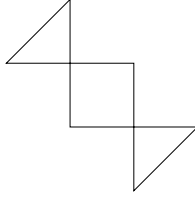
of size comparable to the mesh size of $\mathcal{T}_i$. Here the square has multiplicity $+1$, while the two triangles have multiplicity -1. Thus the *algebraic* area of the figure is zero, in accord with the value $\det D^2f = 0$, but the contribution to the mass of $\mathbb{D}(p_i)$ is twice the area of the square. Adding these contributions over all the vertices of $\mathcal{T}_i$ yields a term on the order of the integral of the last summand of (2.7). The corresponding term in the estimate we want is zero.

Fortunately, if the subdivided square grid is nearly aligned with the eigenvectors of D^2f then this bad behavior does not occur. So we construct such grids locally in regions where the eigenvectors don't vary too much, keeping the different grids separated by a distance proportional to the mesh size but together covering most of U:
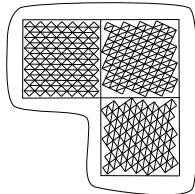
Then we invoke the guarantee in a mesh interpolation algorithm of Chew [2] to conclude that the interstices can be filled in by a triangulation of the same mesh size and uniform fatness. Since the area covered by the interpolated triangles is small, the estimate (2.7) tells us that the price we pay here is not too great.

The primary obstacle to extending this argument to $n \geq 3$ is the absence of a Chew-type algorithm in higher dimensions. From our (superficial) knowledge of the relevant literature this appears to be a fundamental and poorly understood issue in the theory of mesh generation; cf. e.g. [15].

3. FURTHER REMARKS AND QUESTIONS

3.1. **What do these questions have to do with geometric modeling?** Three dimensional modeling was one of the primary motives in the origins of surface theory in the 18th and 19th centuries. Physical objects were supposed to look like smooth domains, once irrelevant irregularities were ignored. The curvature (or the second fundamental form) provided an appealing mathematical tool with serious practical applications.

In the modern era, when computers are widely available and we no longer expect nature to behave necessarily in a smooth regular fashion, the assumption that messy natural formations can be thought of as C^2 smooth seems quaint. In this setting it is desirable to possess a more robust but still natural mathematical model that would nonetheless retain some of the main measurements such as curvature. The normal/gradient cycle of X or f provides such a tool. To put it another way, objects and functions that are regular enough to be associated to such cycles provide a model for what a natural geometric object should look like: the total volume of the cycle gives a gross numerical measure of "total curvedness" of the object, which may be distributed either smoothly or else in some irregular fashion. This tool seems uncannily applicable to physical configurations over a wide range of scales. It is tempting to take the existence of this cycle as a certificate of citizenship in the country of "geometrically valid" objects.

Conjectures (2.5) and (2.6) may be rephrased colloquially as: can we use classical mathematical analysis (C^2) or quasi-discrete computer models (PL) to survey this country to any arbitrarily given degree of accuracy? This would be roughly analogous to some basic facts from integration theory: a given signed Radon measure may be approximated weakly either by discrete or by absolutely continuous measures of the same mass.

3.2. **Towards a proof of** (2.5) **for** $n = 2$**.** Can this method be adapted to construct a strong PL approximation of a general $f \in \mathrm{MA}(\mathbb{R}^2)$?

3.3. **Is there a more natural approach to Theorem 2?** The proof of Theorem 2 in [1] seems somewhat ad hoc. A more natural proof might be possible, based on a certain well known and alluring but almost completely unexplored smoothing strategy.

The basic idea seems to have been mentioned first in [7]: if $X \subset \mathbb{R}^n$ is a compact set, and $r > 0$ is a regular value in the sense of Clarke [3] of the distance function $\delta_X := \mathrm{dist}(\cdot, X)$, then the superlevel set $\overline{X_r} := \{\delta_X \geq r\}$ has positive reach. If such r is small then for $1 \gg r \gg s > 0$ the set $X_{r,s} := \{\delta_{\overline{X_r}} \geq s\}$ is a $C^{1,1}$ domain that is close (with respect to the Hausdorff metric) to X. Furthermore the mass of the normal cycle of $X_{r,s}$ is close to that of $\overline{X_r}$. It is then easy to find a C^2 (or even C^∞) domain close to $X_{r,s}$ whose normal cycle has almost the same mass.

Supposing X to admit a normal cycle in its own right, it is tempting to carry out this procedure to try to construct a smooth domain close to X whose normal cycle has mass close to that of $N(X)$. The missing ingredient is a good estimate for the mass of $N(\overline{X_r})$ in terms of the mass of $N(X)$. In certain tightly circumscribed settings, a weaker kind of estimate is available: if X is subanalytic (e.g. a polyhedron) then the masses of the $N(\overline{X_r})$ are uniformly bounded for small $r > 0$ (this is the basis for the discussion of this subject in [9]). However, the known bound is not geometric in nature, depending instead on the complexity of the description of X as a subanalytic set. Thus it may behave badly with respect to the mass of $N(X)$.

Therefore (passing from the geometric to the functional realm) at present this approach does not yield a proof of Theorem 2. Although for each particular PL function p it yields a sequence of smooth f_i with a uniform bound on the $\mathbb{D}(f_i)$, this bound depends on the complexity of the description of p. For example, if $\mathbb{D}(p)$ is very close to the gradient cycle of a constant function, but p consists of a great many small affine pieces, the known bound on the masses of the approximating $\mathbb{D}(f_i)$ will be very large. No such general bounds in terms of the mass of $\mathbb{D}(p)$ or $N(X)$ in any nontrivial instance have been given in the literature, whether proved or conjectured.

The simplest case is that of a PL function of two variables. For this it would be enough to prove such a bound in the neighborhood of a vertex, or in other words for PL functions that are homogeneous. Since the question is now being phrased in terms of functions, it seems convenient to replace the tube construction $X \mapsto \overline{X_r}$ above by the functional analogue $p \mapsto p_r$, where for each $r > 0$ we put

$$p_r(x) := \sup_y \; p(y) - \frac{1}{r}|y - x|^2$$

Note that each $p_r, r > 0$, is semiconvex. Semiconvexity is the functional analogue of the positive reach condition.

Let $p \in PL(\mathbb{R}^2)$, with $p(tx) = tp(x)$ for $t \geq 0$ and $x \in \mathbb{R}^2$. Is there a universal local bound on the area of $\mathbb{D}(p_r)$ in terms of that of $\mathbb{D}(p)$, valid for r small? By homogeneity it is enough to understand the case $r = 1$.

References

[1] U. Brehm and W. Kühnel. Smooth approximation of polyhedral surfaces regarding curvatures. *Geom. Dedicata*, 12:435–461, 1982.

[2] L.P. Chew. Guaranteed quality triangular meshes., 1989. `http://www.cs.sandia.gov/~samitch/unm_math_579/course_papers/chew_guaranteed_triangular_89-983.pdf`.

[3] F.H. Clarke. *Optimization and nonsmooth analysis*. Wiley, New York, 1990.

[4] D. Cohen-Steiner and J.-M. Morvan. Second fundamental measure of geometric sets and local approximation of curvatures. *J. Differential Geom.*, 74:363–394, 2006.

[5] H. Federer. Curvature measures. *Trans. Amer. Math. Soc.*, 93:418–491, 1959.

[6] H. Federer. *Geometric measure theory*. Springer, New York, 1969.

[7] J.H.G. Fu. Tubular neighborhoods in Euclidean spaces. *Duke Math. J.*, 52:1025–1046, 1985.

[8] J.H.G. Fu. Monge-Ampère functions I, II. *Indiana Univ. Math. J.*, 38:745–771, 773–789, 1989.

[9] J.H.G. Fu. Curvature measures of subanalytic sets. *Amer. J. Math.*, 116:819–880, 1994.

[10] J.H.G. Fu. An extension of Alexandrov's theorem on second derivatives of convex functions. *Adv. Math.*, 228:2258–2267, 2011.

[11] J.H.G. Fu and R.C. Scott. Piecewise linear approximation of smooth functions of two variables. *Adv. Math.*, 248:435–461, 2013.

[12] R. L. Jerrard. Some remarks on Monge-Ampère functions. In *Singularities in PDE and the calculus of variations*, volume 44 of *CRM Proc. Lecture Notes*, pages 89–112. Amer. Math. Soc., Providence, RI, 2008.

[13] R. L. Jerrard. Some rigidity results related to Monge-Ampère functions. *Canad. J. Math.*, 12:320–354, 2010.

[14] D. Pokorný and J. Rataj. Normal cycles and curvature measures of sets with d.c. boundary. *Adv. Math.*, 248:963–985, 2013.

[15] J. Shewchuk. Tetrahedral mesh generation by Delaunay refinement. In *Proceedings of the Fourteenth Annual Symposium on Computational Geometry (Minneapolis, Minnesota)*, pages 86–95, 1998.

[16] M. Zähle. Curvatures and currents for unions of sets with positive reach. *Geom. Dedicata*, 23:155–171, 1987.

Department of Mathematics, University of Georgia, Athens GA 30602, USA • fu@math.uga.edu

Fast numerical schemes related to curvature minimization: a brief and elementary review

Xue-Cheng TAI

Abstract

We will treat variational models that use Euler's elastica and related higher order derivatives as regularizers. These models normally lead to higher order partial differential equations with complicated nonlinearities. It is difficult to solve these equations numerically. Recently, some fast numerical techniques have been proposed that can solve these equations with very good numerical speed. We will try to explain the essential ideas of these numerical techniques and point to some central implementation details for these algorithms.

1. INTRODUCTION

Variational models are becoming essential for image processing and computer vision. A variational model normally needs to minimize an energy functional. This energy functional usually has a "fitting" part and also a "regularizer" part. In this work, we will specially be interested in regularizers involving higher order derivatives.

The goal of image denoising is to remove noise while keeping meaningful vision information such as object edges and boundaries. It is a crucial step in image processing with a wide range of applications in medical image analysis, video monitoring, and others. One of the most popular variational models was proposed by Rudin, Osher, and Fatemi in their seminal work (ROF model) [34]. In [34], a cleaned image is obtained by minimizing the following energy functional

$$(1.1) \qquad E(u) \quad = \quad \int_\Omega |\nabla u| + \frac{\eta}{2} \int_\Omega (f-u)^2,$$

where $f : \Omega \to \mathbb{R}$ is a given noisy image defined on Ω, $\int_\Omega |\nabla u|$ stands for the total variation of a function u (see [40] for a definition), and $\eta > 0$ is a positive tuning parameter controlling how much noise will be removed. The remarkable feature of the ROF model lies in its effectiveness in preserving object edges while removing noise. In fact, the total variation regularizer has been widely employed in accomplishing other image processing tasks such as deblurring, segmentation, and registration.

In order to incorporate more geometrical information into the regularizer, a number of higher order regularization models have been proposed and used for image processing and computer vision problems. In this work, we will mainly consider three higher models and outline some fast algorithms to solve them, c.f. Section 2. To make the presentation clearer, we will only state these models for simple image restoration problems with a given noisy image f. There exist extensions of these models for more general applications related to image processing and computer vision including deblurring, inpainting, zooming and geometry minimization.

2. Higher order regularizations

The ROF model has several unfavorable features. The main caveat is the stair-case effect, that is, the resulting clean image would present blocks even though the desired image could be smooth. Other undesirable properties include corner smearing and loss of image contrast. To remedy these drawbacks, a very rich list of results exists in the literature, see [24, 1, 11, 49, 52]. Despite the effectiveness of these models in removing the staircase effect, it is often a challenging issue to minimize the corresponding functionals. Note that the models contain second order derivatives, the related Euler-Lagrange equations are fourth-order. It is a nontrivial task to develop effective and efficient algorithms to solve these higher order nonlinear equations.

In this section, we will first give an informal introduction to three higher order models. In the subsequent sections, we will introduce fast algorithms to solve them.

Remark 1. We remark on a few important issues for the models and algorithms:

(1) The introduction about the higher order mathematical models is very informal in a mathematical sense. To define and analyze these models for proper function spaces is far more difficult and is beyond the content of this note. Analysis of these models in the continuous setting is still incomplete in the literature.

(2) For image processing problems, the computation domain is always a rectangle. The pixels of an image give a ready mesh for the discretization. This leads to easy discretization with finite difference approximations for our models and algorithms. However, this is not a restriction for the models. The models and algorithms are valid for general domains as well. For general domains with curved boundaries, the discrete approximation of the functions (including primal and auxiliary functions) could be complicated near the boundaries.

(3) In the literature, discrete curvature is often used, see other publications in this proceeding. Instead, we have chosen to present these models in a continuous setting. The algorithms are also presented in the continuous setting. For numerical implementation, we must discretize the continuous functions.

Both approaches ("discrete curvature" or "continuous curvature") have advantages and disadvantages. For many discrete curvature models, it seems that the "discrete curvature" is not converging to the "continuous curvature" when the mesh size goes to zero. For the continuous models, it is easy to see that we are using the curvature. Standard finite difference or finite element approximations would lead to natural approximations for the curvature terms. However, a rigorous proof for this is still missing due to the complexity of the models.

2.1. **Regularization using TV2.** In [24], Lysaker et al. directly incorporated second order derivative information into the image denoising process. They proposed to minimize the following energy functional to denoise an image:

$$(2.1) \qquad E(u) \;\; = \;\; \int_\Omega \sqrt{u_{xx}^2 + u_{xy}^2 + u_{yx}^2 + u_{yy}^2} + \frac{\eta}{2} \int_\Omega (f - u)^2.$$

This higher order energy functional is much simpler than the elastic regularizer that we shall introduce later. Numerically, this regularizer shows rather good performance with noise suppression and edge preservation. In the literature, there exists quite a number of related models, see [13, 20, 42, 3, 12, 6, 18, 36, 31, 5, 14, 8, 19, 48, 30, 7]. The well-posedness for this energy functional and its gradient flow equation have been studied in [27, 26, 17].

2.2. **Regularization using the Euler's Elastica energy.** Given a function $f : \Omega \mapsto R$, the Euler's elastica model needs to find a function u to minimize the following energy functional:

$$(2.2) \qquad E(u) \;\; = \;\; \int_\Omega \left[a + b \left(\nabla \cdot \frac{\nabla u}{|\nabla u|} \right)^2 \right] |\nabla u| + \frac{\eta}{2} \int_\Omega (f - u)^2.$$

The nonnegative constants a, b and η need to be chosen properly for different purposes of applications. This model comes from the Euler's Elastca energy for curves, see [11] for some more explanations for this energy. For a given curve $\Gamma \subset R^2$ with curvature κ, the Euler's elastica

energy is defined as:

$$\int_\Gamma (a + b\kappa^2)ds.$$

For a function u, the curvature for the level curve $\Gamma_c : u(x) = c$ is:

$$\kappa = \nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right).$$

Thus, the Euler's elastica energy for the level curve: $u(x) = c$ is:

$$\ell(c) = \int_{\Gamma_c} \left(a + b\left[\nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right)\right]^2\right)ds.$$

Summing up (integrating) the Euler's elastica energy for all the level curves $\Gamma_c : c \in (-\infty, \infty)$, we get from the co-area formula [39] that the total Euler's elastica energy for all the level curves is:

$$\int_{-\infty}^{\infty} \ell(c)dc = \int_{-\infty}^{\infty} \int_{\Gamma_c} \left(a + b\left[\nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right)\right]^2\right)dsdc = \int_\Omega \left(a + b\left[\nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right)\right]^2\right)|\nabla u|dx.$$

Minimization problem (2.2) is trying to use the total Euler's Elastica energy as a regularizer to remove noise from the image f.

2.3. **Regularization using the image surface mean curvature.** In [49], the authors proposed a variational model that uses the mean curvature of the induced image surface $(x, y, f(x, y))$ to remove noise. Specifically, the model employs the L^1 norm of the mean curvature of the image surface as the regularizer and minimizes the following functional to get a clean image:

$$(2.3) \qquad E(u) = \int_\Omega \left|\nabla \cdot \left(\frac{\nabla u}{\sqrt{1 + |\nabla u|^2}}\right)\right| + \frac{\eta}{2}\int_\Omega (f - u)^2.$$

Above, η is a tuning parameter. The term $\nabla \cdot (\frac{\nabla u}{\sqrt{1+|\nabla u|^2}})$ is the mean curvature of the surface $\phi(x, y, z) = u(x, y) - z = 0$. The model tries to fit the given noisy image surface $(x, y, f(x, y))$ with a surface $(x, y, u(x, y))$ that is regularized by the mean curvature. This idea can be traced to much earlier papers, see [22]. The model can sweep noise while keeping object edges, and it also avoids the staircase effect. More importantly, as discussed in [49, 53], the model is also capable of preserving image contrasts as well as object corners.

3. Fast numerical algorithms based on augmented Lagrangian method (ALM)

In this section, we first show the split-Bregman altorithm of Goldstein-Osher [16] for the ROF model [34]. We then extend this idea for the three higher order models we have introduced in Section 2 to get fast algorithms for them.

3.1. **Split-Bregman for ROF.** In work by Goldstein-Osher [16], fast iterative schemes were proposed and tested for the ROF model. It is one of the most efficient numerical schemes for solving the ROF model. Later, it was observed that the split-Bregman algorithm of Golstein-Osher [16] is equivelent to the Augmented Lagrangian method [38, 41]. Here, we explain the ideas in an "informal" fashion. We will present the ideas in a continuous setting. As stated in Remark 1, to make our statements precise, more precise definitions of the function spaces and the norms need to be given. That is one of the reasons that discrete models have been used to explain these algorithms as in [41]. We will not get into the details related to this kind of technicalities.

Let $p = \nabla u$, then it is easy to see that the ROF model is equivalent to the following constrained minmization problem:

$$(3.1) \qquad \min_{\substack{u,p \\ p=\nabla u}} \int_\Omega |p| + \frac{\eta}{2}|u - f|^2dx.$$

Let us use the Augmented Lagrangian method (ALM) [15] to deal with the constraint $p = \nabla u$. Then we need to find a saddle point for the following Lagrangian functional:

$$(3.2) \qquad L_{rof}(u, p, \lambda) = \int_\Omega |p| + \frac{\eta}{2}|u - f|^2 + \lambda \cdot (p - \nabla u) + \frac{r}{2}|p - \nabla u|^2dx.$$

Above: $u : \Omega \mapsto R$ denotes the image we need to find, $p : \Omega \mapsto R^2$ is a vector valued function related to the gradient of the function u, $\lambda : \Omega \mapsto R^2$ denotes the Lagrangian multiplier. Due to convexity, problem (3.1) has a unique solution (in the discrete setting as well as in a proper space in the continuous setting). If (u^*, p^*) is a global minimizer for (3.1), then there exists a λ^* such that (u^*, p^*, λ^*) is a saddle point for (3.2). We propose to use Algorithm 1 to search for a saddle point for (3.2).

Algorithm 1 Augmented Lagrangian method for the ROF model

Initialization: $\lambda^0 = 0, u^0 = f$; For k=0,1,2,...:

(1) Compute p^{k+1} from :

$$(3.3) \qquad p^{k+1} = \arg\min_q L_{rof}(u^k, q; \lambda^k),$$

(2) Compute u^{k+1} from:

$$(3.4) \qquad u^{k+1} = \arg\min_v L_{rof}(v, p^{k+1}; \lambda^k),$$

(3) Update

$$(3.5) \qquad \lambda^{k+1} = \lambda^k + r(p^{k+1} - \nabla u^{k+1}).$$

(4) Go to the next iteration if not converged.

Minimization subproblem (3.3) has closed-form solutions and thus can be easily computed. Minimization subproblem (3.4) can be solved by FFT (Fast Fourier Transform) or simple Gauss-Seidel iterative solvers to get an approximate solution. See [43, 41] for more details. Theoretically, it is necessary to have sufficiently many iterations between subproblems (3.3) and (3.4). In practice, the above algorithm works well for most of the cases for the ROF model.

It is also easy to extend the above model for vector-valued functions and vector-TV regularization, see [41, p.320] and [33, 32].

3.2. **Split-Bregman for second order Total variation.** Here, we explain how to use the fast algorithm explained in the last section for the regularization model (2.1) related to second order derivatives. The idea follows the work [41].

The essential idea for the fast schemes is to introduce some auxiliary variables and consider the complicated minimization problem as a constrained minimization. We then use splitting ideas to decompose the complicated minimization problem into some simpler minimization problems. Let

$$p = D^2 u = \begin{pmatrix} u_{xx} & u_{xy} \\ u_{yx} & u_{yy} \end{pmatrix}.$$

Thus p is a matrix function defined on Ω, i.e. p is equal to the Hessian matrix of u over Ω. The minimization of the energy functional given in (2.1) is equivalent to:

$$(3.6) \qquad \min_{\substack{u,p \\ p=D^2u}} \int_\Omega |p| + \frac{\eta}{2}|u - f|^2 dx.$$

Above, $|p| = \sqrt{\sum_{i,j} p_{ij}^2}$ stands for the Frobenius norm of the matrix p.

Again, we use the Augmented Lagrangian method (ALM) [15, 41] to deal with the constraint $p = D^2 u$. Then we need to find a saddle point for the following Lagrangian functional:

$$(3.7) \qquad L_{llt}(u, p, \lambda) = \int_\Omega |p| + \frac{\eta}{2}|u - f|^2 + \lambda : (p - D^2 u) + \frac{r}{2}|p - D^2 u|^2 dx.$$

Here $u : \Omega \mapsto R$ denotes the image we need to find, $p : \Omega \mapsto R^4$ is a matrix valued function related to the Hessian of the function u, $\lambda : \Omega \mapsto R^4$ denotes the Lagrangian multiplier. The notation $A : B$ denotes the elementwise inner product of two matrices A and B. We use Algorithm 2 to search for a saddle point for (3.7).

Similar to Algorithm 1, the minimization subproblem (3.8) needs to compute a matrix-valued function and it has closed-form solutions and thus can be easily computed. Minimization subproblem (3.9) gives raise to a linear 4th order equation on a regular mesh. It can be solved by FFT

Algorithm 2 Augmented Lagrangian method for the TV2 model

Initialization: $\lambda^0 = 0, u^0 = f$; *For k=0,1,2,...:*

   (1) Compute p^{k+1} from

(3.8) $$p^{k+1} = \arg\min_q L_{llt}(u^k, q; \lambda^k),$$

   (2) Compute u^{k+1} from:

(3.9) $$u^{k+1} = \arg\min_v L_{llt}(v, p^{k+1}; \lambda^k),$$

   (3) Update

(3.10) $$\lambda^{k+1} = \lambda^k + r(p^{k+1} - D^2 u^{k+1}).$$

   (4) Go to the next iteration if not converged.

(Fast Fourier Transform) or simple Gauss-Seidel iterative solvers to get an approximate solution. See [41, p.324] for more details.

3.3. **Augmented Lagrangian method for Euler's elastica model.** In order to use fast algorithms related to ALM for the minimization of the Euler's elastica given in (2.2), it is necessary to introduce a few more auxiliary functions. The ideas presented here follow the work [37]. The following lemma is easy to prove using Hölder's inequality:

Lemma 2. *Let $n \neq 0$ and $p \neq 0$ be two given vectors. They satisfy*

$$|n| \leq 1, |p| = n \cdot p,$$

if and only if

$$n = \frac{p}{|p|}.$$

Let us define

$$p = \nabla u, \quad n = \frac{p}{|p|}.$$

It is easy to see that the minimization of the Euler's elastica energy (2.2) is equivalent to the following constrained minimization:

(3.11) $$\min_{u,p,n} \int_\Omega \left(a + b(\nabla \cdot n)^2\right)|p| + \frac{\eta}{2}\int_\Omega |u - f|^2$$
$$\text{with } p = \nabla u, \ |p| = n \cdot p, \ |n| \leq 1.$$

The use of n with $|n| \leq 1$ can be viewed as a relaxation. Moreover, the constraint $|n| \leq 1$ is crucial to prevent the unboundedness of n when $p = \mathbf{0}$. Define the characteristic function $\delta_{\mathcal{R}}(\cdot)$ on $\mathcal{R}$ as

(3.12) $$\delta_{\mathcal{R}}(m) = \begin{cases} 0 & m \in \mathcal{R} \equiv \{m \in \mathbf{L}^2(\Omega) \mid |m| \leq 1 \text{ a.e. in } \Omega\}, \\ +\infty & \text{otherwise.} \end{cases}$$

then, the constrained minimization problem (3.11) can be rewritten as:

(3.13) $$\min_{u,p,n} \int_\Omega \left(a + b(\nabla \cdot n)^2\right)|p| + \frac{\eta}{2}\int_\Omega |u - f|^2 + \delta_{\mathcal{R}}(n)$$
$$\text{with } p = \nabla u, \ |p| = n \cdot p,$$

We know that $|n| \leq 1$ in Ω, thus

$$|p| - n \cdot p \geq 0 \quad \forall x \in \Omega.$$

There are two constraints in (3.13). Two different penalizations are used for these two constraints. For constraint $p = \nabla u$, we use the L2-norm for the penalization term; but for $|p| = n \cdot p$, we use

the L1-norm. With this special treatment for the last constraint, the corresponding Lagrangian functional becomes:

(3.14)
$$L_{elas}(u, p, n, \lambda_1, \lambda_2) = \int_{\Omega} \left(a + b(\nabla \cdot n)^2\right)|p| + \frac{\eta}{2}|u - f|^2 dx + \delta_{\mathcal{R}}(n)$$
$$+ \int_{\Omega} \lambda_1 \cdot (p - \nabla u) + \frac{r_1}{2}|p - \nabla u|^2 + \lambda_2(|p| - n \cdot p) + r_2(|p| - n \cdot p)dx.$$

The meanings of the primal and dual variables are listed in the following:

- $u : \Omega \mapsto R$ denotes the image we need to find,
- $p : \Omega \mapsto R^2$ is a vector-valued function related to the gradient of the function u,
- $n : \Omega \mapsto R^2$ is a vector-valued function related to the unit vectors of the level curves of u,
- $\lambda_1 : \Omega \mapsto R^2$ denotes the Lagrangian multiplier for constraint $p = \nabla u$,
- $\lambda_2 : \Omega \mapsto R$ denotes the Lagrangian multiplier for constraint $|p| = n \cdot p$.

We shall use Algorithm 3 to search for a saddle point of this Lagrangian functional.

Algorithm 3 Augmented Lagrangian method for the Euler's elastica model

Initialization: $\lambda^0 = 0, u^0 = f, n^0 = 0$; *For k=0,1,2,...*:

(1) Compute p^{k+1} from

(3.15)
$$p^{k+1} = \arg\min_{q} L_{elas}(u^k, q, n^k; \lambda^k),$$

(2) Compute n^{k+1} from

(3.16)
$$n^{k+1} = \arg\min_{m} L_{elas}(u^k, p^{k+1}, m; \lambda^k),$$

(3) Compute u^{k+1} from:

(3.17)
$$u^{k+1} = \arg\min_{v} L_{elas}(v, p^{k+1}, n^{k+1}; \lambda^k),$$

(4) Update

(3.18)
$$\lambda_1^{k+1} = \lambda_1^k + r_1(p^{k+1} - \nabla u^{k+1}),$$

(3.19)
$$\lambda_2^{k+1} = \lambda_2^k + r_2(|p^{k+1}| - n^{k+1} \cdot p^{k+1}).$$

(5) Go to the next iteration if not converged.

In the following, we give some remarks on the solutions of the subproblems and some implementation issues for Algorithm 3:

(1) Minimization subproblem (3.15) has closed-form solutions. A simple thresholding is sufficient to get the solution p^{k+1}.

(2) Minimization subproblem (3.16) can be approximated by solving a linear partial differential equation first and then projecting the obtained solution onto the convex set $\mathcal{R}$ defined in (3.12). The linear equation is:

(3.20)
$$-b\nabla(|p^{k+1}|\nabla \cdot n^{k+1}) = (\lambda_2 + r_2)p^{k+1}.$$

This equation has some similarities with the gradient-divergence equations from the Maxwell equation for magnetic simulations or Darcy-Stokes flow, see [25]. It can be approximately solved by a few Gauss-Seidel iterations or a coefficient freezing FFT solver, see [37] for more details. Note that both p and n are vector-valued functions. We need to solve for all the components of the vector functions.

(3) The minimizer u^{k+1} of subproblem (3.17) satisfies the following linear partial differential equation:
$$u^{k+1} - f + \nabla \cdot \lambda_1^k + r_1 \nabla \cdot (p^{k+1} - \nabla u^{k+1}) = 0.$$

This equation is the same as for the ROF model, c.f. Algorithm 1. It can be easily solved by FFT or few Gauss-Seidel iterations.

(4) Theoretically, we need sufficiently many iterations between (3.15)-(3.17) to guarantee convergence of the algorithm. In practice, just one iteration, as stated in Algorithm 3, is enough to have convergence of the iterative solutions.

(5) As the energy functional is not convex, we need to choose some of the penalization parameters r_i sufficiently big to obtain convergence of this algorithm. Tuning the parameters r_i is a delicate issue. Fortunately, there exists an easy way to get the correct values for these penalization parameters r_i. We outline the details in section 5.

(6) Another relaxation method for the Euler's Elastica model was proposed in [4]. It solves the problem by means of tractable convex relaxation in higher dimensions.

The algorithm presented in [37] for the minimization of the Euler's elastica energy has one more auxiliary function variable. There, the following constrained minimization was considered:

$$(3.21) \qquad \min_{u,p,n} \int_\Omega \left(a + b(\nabla \cdot n)^2\right)|p| + \frac{\eta}{s}\int_\Omega |u - f|^s + \delta_\mathcal{R}(m)$$
$$\text{with} \quad p = \nabla u, \quad |p| = n \cdot p, \quad n = m,$$

Note that the fidelity term has also been changed and it is powered by s. Normally, the value of s can be chosen as $s = 1$ or $s = 2$ depending on the nature of the noise contained in f. For salt-and-pepper noise, we prefer to choose $s = 1$. For Gaussian noise, we choose $s = 2$. There could exist cases where we need to choose $s \in [1, \infty]$.

In [37], the following Lagrangian functional is used for the above constrained minimization problem (3.21):

(3.22)

$$L_{elas}(u,p,m,n,\lambda_1,\lambda_3,\lambda_3) = \int_\Omega \left(a + b(\nabla \cdot n)^2\right)|p| + \frac{\eta}{s}|u - f|^s dx + \delta_\mathcal{R}(m)$$
$$+ \lambda_1 \cdot (p - \nabla u) + \frac{r_1}{2}|p - \nabla u|^2 + \lambda_2(|p| - n \cdot p) + r_2(|p| - n \cdot p) + \lambda_3 \cdot (m - n) + \frac{r_3}{2}|n - m|^2 dx$$

An algorithm similar to Algorithm 3 can be used to find a saddle point for the above Lagrangian functional. We will not repeat the details. We can see that the minimization subproblem for n does not have the constraint $|n| \leq 1$ and we only need to solve a linear PDE system to get the values of n^{k+1} which can be done by using FFT or a few Gauss-Seidel iterations, c.f. (3.20). The convex constraint is only imposed on m now. The solution of the minimization subproblem for m is in fact just a simple projection to the convex set, see [37] for the details.

It is also possible to use these ideas for a generalized Euler's elastica model with the energy functional modified to be:

$$(3.23) \qquad E(u) = \int_\Omega \left[a + b\left|\left(\nabla \cdot \frac{\nabla u}{|\nabla u|}\right)\right|^{s_1}\right]|\nabla u| + \frac{\eta}{s_2}\int_\Omega |f - u|^{s_2}.$$

In case that $s_1 = 2, s_2 = 1$, we could consider the following splitting:

$$(3.24) \qquad \min_{u,p,n} \int_\Omega \left(a + b(\nabla \cdot n)^2\right)|p| + \eta \int_\Omega |v - f| + \delta_\mathcal{R}(n)$$
$$\text{with} \quad v = u \quad p = \nabla u, \quad |p| = n \cdot p,$$

It is easy to define the corresponding Lagrangian functional and use an alternating minimization scheme to search for its saddle point. The details of the corresponding algorithm will be omitted and all the minimization subproblems can be easily solved or have closed-form solutions, see [37, p.33] for the needed details.

If we consider the case that $s_1 = 1, s_2 = 2$, then it would be better to use the following splitting idea:

$$(3.25) \qquad \min_{u,p,n} \int_\Omega (a + b|q|)|p| + \frac{\eta}{2}\int_\Omega |u - f|^2 + \delta_\mathcal{R}(m)$$
$$\text{with} \quad q = \nabla \cdot n, \quad p = \nabla u, \quad |p| = n \cdot p, \quad n = m.$$

It is easy to define the corresponding Lagrangian functional and use an alternating minimization scheme to search for its saddle point. The details of the corresponding algorithm will be omitted and all the minimization subproblems can be easily computed or have closed-form solutions, see [37, p.33] for the needed details.

We want to emphasise that for the constraint $|p| - m \cdot p = 0$, we use L1-norm for the penalization and it is true that $|p| - m \cdot p \geq 0$ due to the fact that $|m| \leq 1$.

So far, we have explained the splitting techniques needed for the cases $s_1 = 2, s_2 = 1$ and $s_1 = 1, s_2 = 2$. For the case $s_1 = s_2 = 1$, we need to combine the splitting techniques for both cases. These techniques can be easily extended for more general values of s_1 an s_2. In fact, the algorithm given in [37] can deal with general Lp fidelity terms.

3.4. **Augmented Lagrangian method for the mean curvature (MC) model.** The ideas presented in this section follow the work [53]. Let us recall the idea of introducing the mean curvature denoising model. In this model, a $2D$ image $f(x, y)$ is regarded as a surface $(x, y, f(x, y))$ in $\mathbb{R}^3$, c.f. [22, 49]. One thus considers the surface $\phi(x, y, z) = u(x, y) - z = 0$ and the mean curvature $\kappa = \nabla \cdot (\nabla\phi/|\nabla\phi|) = \nabla \cdot (\langle\nabla u, -1\rangle/|\langle\nabla u, -1\rangle|)$. Here and later, $\langle\cdot, \cdot\rangle$ is used to denote the concatenation of vectors. Note that one introduces two variables $\mathbf{p} = \nabla u$ and $\mathbf{n} = \nabla u/|\nabla u|$ to tackle the Euler's elastica for its curvature term $\kappa = \nabla \cdot (\nabla u/|\nabla u|)$. This gives us a hint on how to treat the curvature term in our case, that is, we may introduce a variable $\mathbf{p} = \langle\nabla u, -1\rangle$ instead of $\mathbf{p} = \nabla u$. Accordinlgy, we will also introduce $\mathbf{n} = \langle\nabla u, -1\rangle/|\langle\nabla u, -1\rangle|$.

With constraints
$$\mathbf{p} = \langle\nabla u, -1\rangle, \mathbf{n} = \langle\nabla u, -1\rangle/|\langle\nabla u, -1\rangle|,$$
the MC model (2.3) is then transformed to the following constrained minimization problem:

$$\min_{u,q,\mathbf{n},\mathbf{p}} \left[\lambda \int_\Omega |q| + \frac{1}{2} \int_\Omega (f - u)^2 \right],$$

(3.26) $$\text{with } q = \nabla \cdot \mathbf{n}, \ \mathbf{n} = \frac{\mathbf{p}}{|\mathbf{p}|}, \ \mathbf{p} = \langle\nabla u, -1\rangle.$$

The associated augmented Lagrangian functional is then:

$$
\begin{aligned}
\mathcal{L}(u, q, \mathbf{p}, \mathbf{n}, \mathbf{m}; \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \boldsymbol{\lambda}_4) &= \lambda \int_\Omega |q| + \frac{1}{2} \int_\Omega (f - u)^2 \\
&+ r_1 \int_\Omega (|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) + \int_\Omega \lambda_1(|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) \\
&+ \frac{r_2}{2} \int_\Omega |\mathbf{p} - \langle\nabla u, -1\rangle|^2 + \int_\Omega \boldsymbol{\lambda}_2 \cdot (\mathbf{p} - \langle\nabla u, -1\rangle) \\
&+ \frac{r_3}{2} \int_\Omega (q - \partial_x n_1 - \partial_y n_2)^2 + \int_\Omega \lambda_3(q - \partial_x n_1 - \partial_y n_2) \\
&+ \frac{r_4}{2} \int_\Omega |\mathbf{n} - \mathbf{m}|^2 + \int_\Omega \boldsymbol{\lambda}_4 \cdot (\mathbf{n} - \mathbf{m}) + \delta_\mathcal{R}(\mathbf{m}),
\end{aligned}
$$

(3.27)

where $r_i's$, $i = 1, \cdots, 4$, are the penalization parameters, and $\lambda_1, \lambda_3 : \Omega \mapsto \mathbb{R}$ and $\boldsymbol{\lambda}_2, \boldsymbol{\lambda}_4 : \Omega \mapsto \mathbb{R}^3$ are Lagrange multipliers, and $\mathbf{p}, \mathbf{n}, \mathbf{m} : \Omega \mapsto \mathbb{R}^3$ are vector-valued functions. For the sake of completeness of presentation, we make a few remarks in the following.

Introduction of the variable $\mathbf{m}$ aims to relax variable $\mathbf{n}$ that is defined as $\mathbf{n} = \mathbf{p}/|\mathbf{p}|$. The variable $\mathbf{m}$ is required to lie in the set $\mathcal{R}$ so that the term $|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}$ is always non-negative. As discussed in [37], the benefit of this non-negativeness is that the L^2 penalization is unnecessary. Instead, we use L1-norm for the penalization. As this term is always positive, the penalization term becomes just $|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}$.

As the saddle points of the augmented Lagrangian functional (3.27) correspond to the minimizers of the constrained minimization problem (3.26), one just needs to find the saddle points of (3.27). Similar to algorithms for the Euler's elastica model, we apply an iterative procedure. Specifically, for each variable in (3.27), we fix all the other variables and seek a critical point of the induced functional to update this variable. Once all the variables are updated, the Lagrangian multipliers will also be updated. Then we repeat the process until the variables converge to a steady state. The algorithm is summarized in Algorithm 4.

The sub-minimization problems (3.28)-(3.32) are very easy to solve. We list their corresponding minimization energy functionals in the following:

(3.33) $$\varepsilon_1(u) = \frac{1}{2} \int_\Omega (f - u)^2 + \frac{r_2}{2} \int_\Omega |\mathbf{p} - \langle\nabla u, -1\rangle|^2 + \int_\Omega \boldsymbol{\lambda}_2 \cdot (\mathbf{p} - \langle\nabla u, -1\rangle),$$

(3.34) $$\varepsilon_2(q) = \lambda \int_\Omega |q| + \frac{r_3}{2} \int_\Omega (q - \partial_x n_1 - \partial_y n_2)^2 + \int_\Omega \lambda_3(q - \partial_x n_1 - \partial_y n_2),$$

Algorithm 4 Alternating minimization method for surface mean curvature minimization.

(1) Initialization: u^0, q^0, $\mathbf{p}^0$, $\mathbf{n}^0$, $\mathbf{m}^0$, and λ_1^0, $\boldsymbol{\lambda}_2^0$, λ_3^0, $\boldsymbol{\lambda}_4^0$. For $k \geq 1$, loop over the following two steps:

(2) Compute an approximate minimizer $(u^k, q^k, \mathbf{p}^k, \mathbf{n}^k, \mathbf{m}^k)$ of the augmented Lagrangian functional with the fixed Lagrangian multiplier λ_1^{k-1}, $\boldsymbol{\lambda}_2^{k-1}$, λ_3^{k-1}, $\boldsymbol{\lambda}_4^{k-1}$ from the following minimization problems:

$$(3.28) \qquad u^k = \operatorname{argmin} \mathcal{L}(u, q^{k-1}, \mathbf{p}^{k-1}, \mathbf{m}^{k-1}, \mathbf{n}^{k-1}, \lambda_1^{k-1}, \boldsymbol{\lambda}_2^{k-1}, \lambda_3^{k-1}, {\lambda_4}^{k-1})$$

$$(3.29) \qquad q^k = \operatorname{argmin} \mathcal{L}(u^k, q, \mathbf{p}^{k-1}, \mathbf{m}^{k-1}, \mathbf{n}^{k-1}, \lambda_1^{k-1}, \boldsymbol{\lambda}_2^{k-1}, \lambda_3^{k-1}, {\lambda_4}^{k-1})$$

$$(3.30) \qquad p^k = \operatorname{argmin} \mathcal{L}(u^k, q^k, \mathbf{p}, \mathbf{m}^{k-1}, \mathbf{n}^{k-1}, \lambda_1^{k-1}, \boldsymbol{\lambda}_2^{k-1}, \lambda_3^{k-1}, {\lambda_4}^{k-1})$$

$$(3.31) \qquad m^k = \operatorname{argmin} \mathcal{L}(u^k, q^k, \mathbf{p}^k, \mathbf{m}, \mathbf{n}^{k-1}, \lambda_1^{k-1}, \boldsymbol{\lambda}_2^{k-1}, \lambda_3^{k-1}, {\lambda_4}^{k-1})$$

$$(3.32) \qquad n^k = \operatorname{argmin} \mathcal{L}(u^k, q^k, \mathbf{p}^k, \mathbf{m}^k, \mathbf{n}, \lambda_1^{k-1}, \boldsymbol{\lambda}_2^{k-1}, \lambda_3^{k-1}, {\lambda_4}^{k-1})$$

(3) Update the Lagrangian multipliers

$$\begin{aligned}
\lambda_1^k &= \lambda_1^{k-1} + r_1(|\mathbf{p}^k| - \mathbf{p}^k \cdot \mathbf{m}^k) \\
\boldsymbol{\lambda}_2^k &= \boldsymbol{\lambda}_2^{k-1} + r_2(|\mathbf{p}|^k - \langle \nabla u^k, -1 \rangle) \\
\lambda_3^k &= \lambda_3^{k-1} + r_3(q^k - \partial_x n_1^k - \partial_y n_2^k) \\
\boldsymbol{\lambda}_4^k &= \boldsymbol{\lambda}_4^{k-1} + r_4(\mathbf{n}^k - \mathbf{m}^k),
\end{aligned}$$

where $\mathbf{n} = \langle n_1, n_2, n_3 \rangle$.

(4) Stop if the given stopping criteria have been satisfied. Otherwise, go to the next iteration.

$$\begin{aligned}
\varepsilon_3(\mathbf{p}) &= r_1 \int_\Omega (|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) + \int_\Omega \lambda_1(|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) + \frac{r_2}{2} \int_\Omega |\mathbf{p} - \langle \nabla u, -1 \rangle|^2 \\
(3.35) &\qquad + \int_\Omega \boldsymbol{\lambda}_2 \cdot (\mathbf{p} - \langle \nabla u, -1 \rangle),
\end{aligned}$$

$$\begin{aligned}
\varepsilon_4(\mathbf{m}) &= r_1 \int_\Omega (|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) + \int_\Omega \lambda_1(|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) + \frac{r_4}{2} \int_\Omega |\mathbf{n} - \mathbf{m}|^2 \\
(3.36) &\qquad + \int_\Omega \boldsymbol{\lambda}_4 \cdot (\mathbf{n} - \mathbf{m}) + \delta_{\mathcal{R}}(\mathbf{m}),
\end{aligned}$$

$$\begin{aligned}
\varepsilon_5(\mathbf{n}) &= \frac{r_3}{2} \int_\Omega (q - \partial_x n_1 - \partial_y n_2)^2 + \int_\Omega \lambda_3(q - \partial_x n_1 - \partial_y n_2) + \frac{r_4}{2} \int_\Omega |\mathbf{n} - \mathbf{m}|^2 \\
(3.37) &\qquad + \int_\Omega \boldsymbol{\lambda}_4 \cdot (\mathbf{n} - \mathbf{m}).
\end{aligned}$$

Fast solvers and closed-form solutions are available for all these subproblems, c.f. [53].

4. Euler's elastica regularizer for interface problems

The classical snake and active contour model was given by Kass, Witkin, and Terzopoulos [21] where they proposed minimizing the functional

$$(4.1) \qquad \mathbf{E}(\mathcal{C}) = \alpha \int_0^1 |\mathcal{C}'(s)|^2 ds + \beta \int_0^1 |\mathcal{C}''(s)| ds - \eta \int_0^1 |\nabla f(\mathcal{C}(s))|^2 ds,$$

where $f : \Omega \to \mathbb{R}$ denotes a given image and $\mathcal{C}(s) : [0,1] \to \Omega$ is a parameterized curve and α, β, and η are some positive tuning parameters. The first two terms impose regularity restriction on the contour while the third one denotes the drive induced by the given image. As the image f has large gradient along object boundaries, the functional $E(\mathcal{C})$ will take a small value when the active contour $\mathcal{C}$ resides on these boundaries.

Mumford and Shah [29] proposed minimizing the following functional:

$$(4.2) \qquad \mathbf{E}(u, K) = \int_{\Omega \setminus K} |\nabla u|^2 dx + \eta \int_\Omega (u - f)^2 dx + \mu \text{Length}(K)$$

with respect to both the function u defined on Ω and the boundary $K \subset \Omega$. η, μ are positive tuning parameters.

The segmentation model of Chan-Vese [9] can be expressed as the minimization of the following functional:

$$(4.3) \qquad \mathbf{E}_{CV}(\phi, c_1, c_2) \quad = \quad \int_\Omega \mu(f - c_1)^2 H(\phi) + (f - c_2)^2 (1 - H(\phi)) + \eta \int_\Omega |\nabla H(\phi)|,$$

where ϕ is a level set function whose zero level curve presents the segmentation boundary, $H(\cdot)$ is the Heaviside function, c_1, c_2 are two scalars, and μ, η are positive parameters. The parameter μ is often set to be 1 in many applications. If the minimizer of the objective functional in the Mumford-Shah's model is restricted to be $u = c_1 H(\phi) + c_2(1 - H(\phi))$, a "binary image", one can easily get Chan-Vese's model.

In Chan-Vese's model, the first two terms are the fitting terms while the third one represents the length of the segmentation boundary. As discussed in [29], the length term prohibits the excessive segmentation boundaries obtained by the Chan-Vese model. Moreover, it also imposes regularity on the boundaries. Chan-Vese model has proven to be an effective segmentation model. However, the length regularization term is insufficient to accomplish the segmentation task under some circumstances. For instance, as shown in Figure 4.1(A), parts of the letters "UCLA" are erased. Even though one can easily recognize the four letters, existing segmentation models, such as Chan-Vese's model, might often capture the existing boundaries instead of restoring the missing ones as illustrated in Figure 4.1(B). In inpainting problems [11], missing image information is also recovered but within given regions assigned in advance. In contrast, we intend to have a segmentation model that can interpolate the missing boundaries automatically without specifying the regions.

(A)   (B)
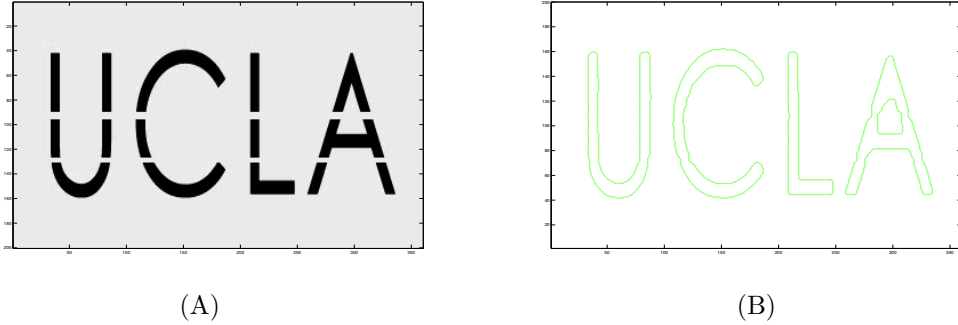
Figure 4.1: Incomplete letters "UCLA" and its integrate segmentation.

To this end, we employ Euler's elastica as a new regularization for the segmentation contour to replace the length term in Chan-Vese's model and get the following functional:

$$\mathbf{E}_{CVE}(\phi, c_1, c_2) \quad = \quad \int_\Omega \mu(f - c_1)^2 H(\phi) + (f - c_2)^2 (1 - H(\phi))$$

$$(4.4) \qquad\qquad\qquad + \left[a + b \left(\nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} \right)^2 \right] |\nabla H(\phi)|,$$

where μ, a, b are positive parameters. For ϕ being the signed distance level set function, it can be proven that the last term equals to the Euler's elastica energy of the segmentation curve. Specifically, the parameter μ has a more important role in this modified model than in Chan-Vese's model. It can relax the competition of the fitting term and the Euler's elastica term, aiming to complete missing boundaries as shown in Figure 4.1(B). The parameters a, b control the length and curvature of segmentation boundary. This regularization was originally proposed and used in the famous work of segmentation with depth by Nitzberg, Mumford, and Shiota [28]. It has also been used in the inpainting problem [11] and the illusory contour problem [51, 50]. Recently, in [35], Schoenemann et al. developed a numerical method to minimize the curvature dependent functionals by using linear programming method. In their work, they also considered Chan-Vese's

model with the substitution of Euler's elastica for the length term. In this section, we shall show the details on using the techniques developed in the earlier sections to minimize the Euler's elastic energy for the CVE model (4.4).

In the current work, we use the same technique as in Section 3.3 to deal with the curvature term in the functional (4.4). Note that the functional (4.4) involves the level set function ϕ, however, only the sign of this function, $H(\phi)$, is needed for the segmentation problem. Following the ideas of the binary level set representation of [23], we introduce a new function $u = H(\phi)$. This was also used in [10] for finding the global minimizer of Chan-Vese's model. More general binary level set representations with global minimization techniques have been developed [45, 47, 44, 2, 46] through some beautiful connections between graph cuts, binary labeling and continuous max-flow problems. As $\nabla \cdot \frac{\nabla H(\phi)}{|\nabla H(\phi)|} = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}$, one can rewrite the functional (4.4) to be

$$(4.5) \quad \mathbf{E}(u, c_1, c_2) = \int_\Omega \mu(f - c_1)^2 u + (f - c_2)^2 (1 - u) + \left[a + b \left(\nabla \cdot \frac{\nabla u}{|\nabla u|} \right)^2 \right] |\nabla u|,$$

where u is supposed to take on either 0 or 1. But note that the curvature makes sense for smooth functions. To fix this issue, as in [2, 10], one can relax the restriction on u to be $0 \le u \le 1$. To minimize the functional (4.5), one considers the following constrained minimization problem

$$\min_{u, \mathbf{p}, \mathbf{n}, c_1, c_2} \int_\Omega \mu(f - c_1)^2 u + (f - c_2)^2 (1 - u) + \left[a + b \left(\nabla \cdot \mathbf{n} \right)^2 \right] |\mathbf{p}|,$$

$$(4.6) \quad \text{with } \mathbf{p} = \nabla u, |\mathbf{p}| = \mathbf{p} \cdot \mathbf{n}, |\mathbf{n}| \le 1, u \in [0, 1].$$

We then construct the following augmented Lagrangian functional:

$$
\begin{aligned}
\mathcal{L}(v, u, \mathbf{p}, \mathbf{n}, \mathbf{m}, c_1, c_2; \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \boldsymbol{\lambda}_4) \quad = \quad & \int_\Omega \mu(f - c_1)^2 v + (f - c_2)^2 (1 - v) + \left[a + b \left(\nabla \cdot \mathbf{n} \right)^2 \right] |\mathbf{p}| \\
+ \quad & r_1 \int_\Omega (|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) + \int_\Omega \lambda_1 (|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m}) \\
+ \quad & \frac{r_2}{2} \int_\Omega |\mathbf{p} - \nabla u|^2 + \int_\Omega \boldsymbol{\lambda}_2 \cdot (\mathbf{p} - \nabla u) \\
+ \quad & \frac{r_3}{2} \int_\Omega (v - u)^2 + \int_\Omega \lambda_3 (v - u) + \delta_\mathcal{D}(v) \\
(4.7) \quad + \quad & \frac{r_4}{2} \int_\Omega |\mathbf{n} - \mathbf{m}|^2 + \int_\Omega \boldsymbol{\lambda}_4 \cdot (\mathbf{n} - \mathbf{m}) + \delta_\mathcal{R}(\mathbf{m}),
\end{aligned}
$$

where $\mathcal{D} = [0, 1]$ and $\mathcal{R} = \{\mathbf{m} \in L^2(\Omega) : |\mathbf{m}| \le 1 \text{ a.e. in } \Omega\}$, and $\delta_\mathcal{D}(v)$ and $\delta_\mathcal{R}(\cdot)$ are the characteristic functions on the sets $\mathcal{D}$ and $\mathcal{R}$ respectively:

$$
\delta_\mathcal{D}(v) = \begin{cases} 0, & v \in \mathcal{D}; \\ +\infty, & \text{otherwise.} \end{cases}
$$

$$
\delta_\mathcal{R}(\mathbf{m}) = \begin{cases} 0, & \mathbf{m} \in \mathcal{R}; \\ +\infty, & \text{otherwise.} \end{cases}
$$

Moreover, $r_i, i = 1, ..., 4$ are positive parameters while $\lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \boldsymbol{\lambda}_4$ are Lagrange multipliers. In this augmented Lagrangian functional, as was explained in Section 3.3, the new variable $\mathbf{m}$ is introduced to simplify the associated subproblem on $\mathbf{p}$. As $\mathbf{m}$ is required to be inside $\mathcal{R}$, $|\mathbf{m}| \le 1$, then $|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m} \ge 0$ for any $\mathbf{p}$, and $|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m} = 0$ if and only if $\mathbf{m} = \frac{\mathbf{p}}{|\mathbf{p}|}$. This avoids the term $\int_\Omega (|\mathbf{p}| - \mathbf{p} \cdot \mathbf{m})^2$, which results in a relatively complex functional on $\mathbf{p}$. Moreover, by using the new variable $\mathbf{m}$, the minimizer of the functional related to $\mathbf{p}$ can be obtained exactly and explicitly by using some appropriate shrinkage.

It is well known that some saddle point of the augmented Lagrangian functional (4.7) relates to a minimizer of the functional (4.5). Therefore, one just needs to find the saddle points of the augmented functional. The minimization energy functional for the CVE model given in (4.6) is very similar to the Euler's elastica energy of Section 3.3. We could use an alternating minimization procedure to approximately minimize the variables $u, v, \mathbf{p}, \mathbf{n}, c_1, c_2$ and use a simple gradient ascent method to update the Lagrange multipliers. Algorithms and numerical performance are exposed in [54].

5. Tuning of the penalization parameters for ALM

The values of the penalization parameters r_i are very important. They influence the convergence as well as the speed of convergence of the proposed algorithms. Fortunately, there exist some very easy techniques to find the proper intervals for the values of these penalization parameters. Here we review some details.

The ROF model associated with the energy functional (3.1) is convex, thus the ALM is convergent for any positive values of r_i used in Algorithm 1. However, the speed of the convergence depends on the values of r_i. Choosing r_i too big or too small could result in more iterations for the solution to converge to the same stopping cretria. For the other higher order models discussed earlier, the values of r_i also influence speed of convergence. In addition, for non-covex energy functionals, some of the penalization parameters need to be sufficiently large to guarantee the convergence.

Fortunately, there are good numerical indicators to use for the determination of the values of r_i. This makes it very easy to tune the penalization parameters. The indicators are related to the constraint errors and the decay of the energy functional value. Let us take Algorithm 4 as an example. First, we need to monitor the constraint errors:

$$(5.1) \quad (R_1^k, R_2^k, R_3^k, R_4^k) = (\|R_1^k\|_{L^1}/\|R_1^0\|_{L^1}, \|R_2^k\|_{L^1}/\|R_2^0\|_{L^1}, \|R_3^k\|_{L^1}/\|R_3^0\|_{L^1}, \|R_4^k\|_{L^1}/\|R_4^0\|_{L^1}),$$

with

$$\begin{aligned}
R_1^k &= |\mathbf{p}^k| - \mathbf{p}^k \cdot \mathbf{m}^k, \\
R_2^k &= \mathbf{p}^k - \langle \nabla u^k, 1 \rangle, \\
R_3^k &= q^k - \partial_x n_1^k - \partial_y n_2^k, \\
R_4^k &= \mathbf{n}^k - \mathbf{m}^k,
\end{aligned}$$

Note that all the errors are normalized by scaling the errors with their values from the first iteration. In addition, we also need to monitor the value of the energy functional. Here are some "troubleshooting" tips on how to tune the parameters r_i:

Step 1 Take some reasonable guess for the values of all the r_i and run the algorithms until the stopping criteria are satisfied.

Step 2 Tune the values of r_i so that the constraint errors R_i^k converge to zero with nearly the same speed asymptotically. If R_i^k goes to zero slower than the others, then increase the value of r_i. If R_i^k goes to zero quicker than the others, then decrease the value of r_i. It is possible that these constraints errors "behave" rather chaotically in the starting phase of the iterations. However, they shall converge to zero asymptotically with the same "speed" if the values of the r_i are chosen correctly.

Step 3 By choosing the penalization values r_i sufficiently large, it is always possible to make the constraint errors go to zero. However, the energy functional value may stay large all the time. For ALM, it is not possible to guarantee that the energy functional will decrease monotonically. However, the energy will decrease and then stay at a constant value if r_i are chosen correctly. Thus, if the constraint errors are decreasing correctly, but not the energy functional value, then reduce all the r_i and repeat this tuning process from step 2.

We also need to stop the iterations properly. In all our numerical experiments, we use the relative residuals (5.1), the relative errors of Lagrange multipliers and value of $E(u^k)$ as the stopping criteria. To check the convergence of the iteration process, we first check on R_i^k. As in [37], we also check the relative errors of Lagrange multipliers:

$$(L_1^k, L_2^k, L_3^k, L_4^k) = \left(\frac{\|\lambda_1^k - \lambda_1^{k-1}\|_{L^1}}{\|\lambda_1^{k-1}\|_{L^1}}, \frac{\|\boldsymbol{\lambda}_2^k - \boldsymbol{\lambda}_2^{k-1}\|_{L^1}}{\|\boldsymbol{\lambda}_2^{k-1}\|_{L^1}}, \frac{\|\lambda_3^k - \lambda_3^{k-1}\|_{L^1}}{\|\lambda_3^{k-1}\|_{L^1}}, \frac{\|\boldsymbol{\lambda}_4^k - \boldsymbol{\lambda}_4^{k-1}\|_{L^1}}{\|\boldsymbol{\lambda}_4^{k-1}\|_{L^1}} \right),$$

(5.2)

and the relative error of the solution u^k

$$(5.3) \quad \frac{\|u^k - u^{k-1}\|_{L^1}}{\|u^{k-1}\|_{L^1}}.$$

Besides all these quantities, we also consider how the energy (2.3) is evolving during the iterations by tracking the value of $E(u^k)$. If all the residual errors R_i^k satisfy the stopping criteria $R_i^k < \epsilon_r$ for some given small threshold ϵ_r, the relative errors for the multipliers and the solution u have been reduced to a sufficiently small level (normally can be close to machine accuracy) and the energy functional $E(u^k)$ has come to a steady constant value, then the algorithm has reached a steady state and we can stop the iterations.

References

[1] Luigi Ambrosio and Simon Masnou. A direct variational approach to a problem arising in image reconstruction. *Interfaces and Free Boundaries*, 5(1):63–82, 2003.

[2] Egil Bae, Jing Yuan, and Xue-Cheng Tai. Global minimization for continuous multiphase partitioning problems using a dual approach. *International Journal of Computer Vision*, pages 1–18, 2010.

[3] Kristian Bredies. Recovering piecewise smooth multichannel images by minimization of convex functionals with total generalized variation penalty. *SFB Report*, 6, 2012.

[4] Kristian Bredies, Thomas Pock, and Benedikt Wirth. Convex relaxation of a class of vertex penalizing functionals. *Journal of mathematical imaging and vision*, 47(3):278–302, 2013.

[5] Carlos Brito-Loeza and Ke Chen. On high-order denoising models and fast algorithms for vector-valued images. *Image Processing, IEEE Transactions on*, 19(6):1518–1527, 2010.

[6] Jeff Calder, A Mansouri, and Anthony Yezzi. Image sharpening via sobolev gradient flows. *SIAM Journal on Imaging Sciences*, 3(4):981–1014, 2010.

[7] Antonin Chambolle and Pierre-Louis Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.

[8] RH Chan, A Lanza, S Morigi, and F Sgallari. An adaptive strategy for the restoration of textured images using fractional order regularization. *Numerical Mathematics: Theory, Methods & Applications*, 6(1), 2013.

[9] T. Chan and L.A. Vese. Active contours without edges. *IEEE Trans Image Proc.*, 10:266–277, 2001.

[10] Tony F. Chan, Selim Esedoglu, and Mila Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM J. Appl. Math.*, 66(5):1632–1648 (electronic), 2006.

[11] Tony F Chan, Sung Ha Kang, and Jianhong Shen. Euler's elastica and curvature-based inpainting. *SIAM Journal on Applied Mathematics*, pages 564–592, 2002.

[12] Eduardo Cuesta, Mokhtar Kirane, and Salman A Malik. Image structure preserving denoising using generalized fractional time integrals. *Signal Processing*, 92(2):553–563, 2012.

[13] Stephan Didas, Joachim Weickert, and Bernhard Burgeth. Properties of higher order nonlinear diffusion filtering. *Journal of mathematical imaging and vision*, 35(3):208–226, 2009.

[14] Yuping Duan and Weimin Huang. A fixed-point augmented lagrangian method for total variation minimization problems. *Journal of Visual Communication and Image Representation*, 24(7):1168–1181, 2013.

[15] R. Glowinski and P. Le Tallec. *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*, volume 9. Society for Industrial Mathematics, 1989.

[16] T. Goldstein and S. Osher. The split bregman method for l1 regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[17] John B Greer and Andrea L Bertozzi. Traveling wave solutions of fourth order pdes for image processing. *SIAM Journal on Mathematical Analysis*, 36(1):38–68, 2004.

[18] Patrick Guidotti and Kate Longo. Two enhanced fourth order diffusion models for image denoising. *Journal of Mathematical Imaging and Vision*, 40(2):188–198, 2011.

[19] Langhua Hu, Duan Chen, and Guo-Wei Wei. High-order fractional partial differential equation transform for molecular surface construction. *Molecular based mathematical biology*, 1:1–25, 2013.

[20] P Jidesh and Santhosh George. Fourth–order variational model with local–constraints for denoising images with textures. *International Journal of Computational Vision and Robotics*, 2(4):330–340, 2011.

[21] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[22] Ron Kimmel, Ravi Malladi, and Nir Sochen. Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images. *International Journal of Computer Vision*, 39(2):111–129, 2000.

[23] Johan Lie, Marius Lysaker, and Xue Tai. A binary level set model and some applications to mumford-shah image segmentation. *IEEE Transactions on Image Processing*, 15(5):1171–1181, 2006.

[24] M. Lysaker, A. Lundervold, and X.C. Tai. Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. *Image Processing, IEEE Transactions on*, 12(12):1579–1590, 2003.

[25] Kent Andre Mardal, Xue-Cheng Tai, and Ragnar Winther. A robust finite element method for darcy-stokes flow. *SIAM Journal on Numerical Analysis*, pages 1605–1631, 2003.

[26] Lihua Min, Xiaoping Yang, and Changfeng Gui. Entropy estimates and large-time behavior of solutions to a fourth-order nonlinear degenerate equation. *Communications in Contemporary Mathematics*, 15(04), 2013.

[27] Lihua Min, Xiaoping Yang, and Dong Ye. Well-posedness for a fourth order nonlinear equation related to image processing. *Nonlinear Analysis: Real World Applications*, 2013.

[28] D Mumford, M Nitzberg, and T Shiota. Filtering, segmentation and depth. *Lecture Notes in Computer Science*, 662, 1993.

[29] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math, 42*, 42:577–685, 1989.

[30] Ehsan Nadernejad and Søren Forchhammer. Wavelet-based image enhancement using fourth order pde. In *Intelligent Signal Processing (WISP), 2011 IEEE 7th International Symposium on*, pages 1–6. IEEE, 2011.

[31] Carola-Bibiane Papafitsorosand Schönlieb. A combined first and second order variational approach for image reconstruction. *Journal of Mathematical Imaging and Vision*, 48(2):308–338, 2014.

[32] Guy Rosman, Alex M Bronstein, Michael M Bronstein, Xue-Cheng Tai, and Ron Kimmel. Group-valued regularization for analysis of articulated motion. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 52–62. Springer, 2012.

[33] Guy Rosman, Yu Wang, Xue-Cheng Tai, Ron Kimmel, and Alfred M Bruckstein. Fast regularization of matrix-valued images. In *Computer Vision–ECCV 2012*, pages 173–186. Springer, 2012.

[34] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[35] T. Schoenemann, F. Kahl, and D. Cremers. Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 17–23. IEEE, 2009.

[36] Carola-bibiane Schönlieb and Andrea Bertozzi. Unconditionally stable schemes for higher order inpainting. *Communications in Mathematical Sciences*, 9(2):413–457, 2011.

[37] X.C. Tai, J. Hahn, and G.J. Chung. A fast algorithm for euler's elastica model using augmented lagrangian method. *SIAM Journal on Imaging Sciences*, 4:313, 2011.

[38] Xue-Cheng Tai and Chunlin Wu. Augmented lagrangian method, dual methods and split bregman iteration for rof model. *Scale Space and Variational Methods in Computer Vision*, pages 502–513, 2009.

[39] Wekipedia. Co-area formula (http://en.wikipedia.org/wiki/coarea_formula), 2013.

[40] Wikipedia. Total variation (http://en.wikipedia.org/wiki/total_variation), 2014.

[41] C. Wu and X.C. Tai. Augmented lagrangian method, dual methods, and split bregman iteration for rof, vectorial tv, and high order models. *SIAM Journal on Imaging Sciences*, 3(3):300–339, 2010.

[42] Fenlin Yang, Ke Chen, and Bo Yu. Efficient homotopy solution and a convex combination of rof and llt models for image restoration. *International Journal of Numerical Analysis & Modeling*, 9(4), 2012.

[43] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for \ell_1-minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.

[44] J. Yuan, E. Bae, and X.C. Tai. A study on continuous max-flow and min-cut approaches. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2217–2224. IEEE, 2010.

[45] J. Yuan, E. Bae, X.C. Tai, and Y. Boykov. A continuous max-flow approach to potts model. *Computer Vision–ECCV 2010*, pages 379–392, 2010.

[46] Jing Yuan, Egil Bae, Xue-Cheng Tai, and Yuri Boykov. A spatially continuous max-flow and min-cut framework for binary labeling problems. *Numerische Mathematik*, pages 1–29, 2013.

[47] Jing Yuan, Juan Shi, and Xue-Cheng Tai. A convex and exact approach to discrete constrained tv-l1 image approximation. *East Asian Journal on Applied Mathematics*, to appear.

[48] Weili Zeng, Xiaobo Lu, and Xianghua Tan. Non-linear fourth-order telegraph-diffusion equation for noise removal. *IET Image Processing*, 7(4):335–342, 2013.

[49] W. Zhu and T. Chan. Image denoising using mean curvature of image surface. *SIAM Journal on Imaging Sciences*, 5(1):1–32, 2012.

[50] W. Zhu, T. Chan, and Selim Esedoglu. Segmentation with depth: A level set approach. *SIAM Journal on Scientific Computing*, 28(5):1957–1973, 2006.

[51] Wei Zhu and Tony Chan. A variational model for capturing illusory contours using curvature. *Journal of Mathematical Imaging and Vision*, 27(1):29–40, 2007.

[52] Wei Zhu, Xue-Cheng Tai, and Tony Chan. Augmented lagrangian method for a mean curvature based image denoising model. *Inverse Problems and Imaging*, 7(3):1075–1097, 2012.

[53] Wei Zhu, Xue-Cheng Tai, and Tony Chan. Augmented lagrangian method for a mean curvature based image denoising model. *Inverse Problems and Imaging*, 7(4):1409–1432, 2013.

[54] Wei Zhu, Xue-Cheng Tai, and Tony Chan. Image segmentation using euler's elastica as the regularization. *Journal of Scientific Computing*, 57(2):414–438, 2013.

Department of Mathematics , University of Bergen, Bergen, Norway • Tai@math.uib.no

Application of discrete curvatures
to surface mesh simplification
and feature line extraction

Alexandra Bac, Jean-Luc Mari, Dimitri Kudelski, Nam-Van Tran, Sophie Viseur,
and Marc Daniel

Abstract

We present two applications of discrete curvatures for surface mesh processing. The first one deals withÊsimplifying a mesh while preserving its sharp features. The second application can be considered as a dual problem, as we investigate ways to detect feature lines within a mesh. Both applications are illustrated with valuable results.

1. Introduction

Estimating shape of discrete objets known by a triangular approximating mesh or even by a point cloud is a relevant problem in the numerous software handling 3D objets. The problem has rather old origins, since one finds its first elements in the works of Gauss and Legendre. The first recent work on the subject was proposed by Alexandrov ([2]). Shape analysis are based on discrete curvature computations and different approaches exist to obtain these second order estimators (see for example [1] for the description of some estimators and results about convergence published in the literature).

We present in this paper two disconnected applications of discrete curvatures for surface mesh processing to illustrate the wide range of information which can be received from theses estimators. The first one deals withÊsimplifying a mesh while preserving its sharp features. Through the quadratic error metric introduced by Garland et al.,Êsuch a simplification can be performed by an edge collapse process guided by the metric. Such an approach leads to highÊquality simplification but remains slow and costly both in terms of space and time. We introduce a two-step method in whichÊwe perform an initial adaptive cell segmentation guided by the curvature and direction of each cell (computed by PCA).ÊThis pre-segmentation according to local curvatures preserves the quality of simplified meshes while reducing computingÊtime by a factor 3 to 4. The second application can be considered as a dual problem, as we investigate ways to detect feature lines within a mesh. Robust extraction of the feature lines of aÊ3D surface model is a challenging problem. Classical approaches generally rely on curvature derivatives, leading to the detection of a salient part as multipleÊsegments despite the fact that it visually appears as a single and fully connected element. We propose a two-step method aiming at extracting feature lines onÊ3D meshes with connectivity preservation. First, all the mesh vertices are labeled according to their curvature values in order to construct regions of interestÊon the discrete surface. The second step consists in a skeletonization directly on the mesh that corresponds to a homotopic thinning of the previouslyÊbinarized areas. Consequently, the resulting lines are highly connected due to the topological properties of the thinning operator.

2. Surface Mesh Segmentation

2.1. **Context.** Our work originates in the study of triangular mesh surfaces originated from geology and geologic surface modelling (as part of a collaboration with the IFP - French Institute of Petroleum). Our data, obtained by physical measures, are typically inhomogeneous, sparse, noisy and voluminous. Therefore, we are interested in the improvement of such surfaces and more particularly in the detection and filling of holes and faults. However, most improvement algorithms are both time and space consuming and thus, it is fundamental to simplify, smooth and homogenize data before any further treatment while preserving curvatures and critical areas such as faults (see [4]).

The present work was undertaken in this context: our hybrid mesh simplification method allies both vertex clustering and iterative edge collapse techniques. These approaches are actually complementary: iterative edge contraction (based on quadratic error metrics, see [10], ([11]), compared to vertex clustering approaches, leads to results of good quality but proves very costly both in terms of time and space. Vertex clustering algorithms (see for instance [20], [7], [17]) are simple, light and efficient methods but they hardly take into account the local geometry of the surface. Therefore, our idea was to combine both an adaptive segmentation step followed by an iterative edge collapse process (this last step ends when the expected simplification rate is reached).

The paper is organized as follows: in section 2.2, we introduce our two step method, while sections 2.3 and 2.4 respectively detail each step. Section 3.4 emphasizes the very interesting results we obtained.

2.2. **Method General presentation.** Our work starts from an observation: the approaches to triangular mesh simplification are various and actually each of them is relevant in its own field. On the one hand, vertex clustering approaches are particularly interesting in terms of time and space consumption and will be more efficient for low simplification rates. On the other hand, iterative edge contraction is slower and requires more memory, but produces better results (specially for high simplification rates).

The purpose of our algorithm is to conciliate the advantages of both approaches in order to efficiently handle models of any size while preserving the quality of the resulting approximations.

The underlying idea of our algorithm is to combine a first adaptive segmentation step with a second iterative edge collapse step.

2.2.1. *Vertex grouping: spatial adaptive clustering.* The first step of our algorithm consists in a vertices grouping step. As we have explained previously, in order to obtain satisfactory results, it is necessary to take into account the local geometry of the surface and hence to use an adaptive approach. However, if the original data is inhomogeneous and if some areas of the original surface are sparse, a purely adaptive approach can lose too many informations in these areas. Therefore, in order to avoid such problems, our algorithm starts from a rough regular grid. This initial grid is then refined by successive approximations: splitting planes are determined by a principal component analysis and inserted in the cells where more detail is necessary (see section 2.3.2).

In order to split cells efficiently, it is necessary to define a priority for their treatment. We chose to estimate the absolute curvature at each vertex (we use the estimation by Meyer et al. [19], see section 2.3.1). The indicator attached to a cell is the sum of the absolute curvatures of its vertices; cells are processed according to this indicator.

Last, a representative vertex is computed for each cell (by minimization of the quadratic error metric associated to the cell), and a topology is rebuilt over these vertices, inherited from the initial topology (see section 2.3.2).

2.2.2. *Iterative edge collapse.* Starting from the intermediate approximation of the mesh obtained by vertices grouping together with the quadratic error matrices previously computed, an iterative edge collapse process is applied in order to produce a smaller and smoother simplification (see section 2.4).

2.3. **Vertex clustering : adaptive segmentation.**

2.3.1. *Discrete curvatures.* A triangular mesh is a piecewise linear surface. Therefore, its curvature (in the sense of differential geometry) is null everywhere except on the edges where it is not defined. However, it can be interesting to consider such a surface as a discrete approximation of a continuous surface. In this perspective, one can define discrete curvature indicators; ideally these discrete indicators should converge to the continuous ones as the mesh density increases. Several definitions have been proposed for such discrete curvature indicators (see [8], [19], [24]). We chose to use the definition by M. Meyer and al. ([19]) as it constitutes a good trade-off between quality and complexity (convergence results have been formally obtained by G. Xu in [26]).

For any vertex v, we use Meyer's estimates to compute both mean curvature H and Gaussian curvature K at v. Let κ_1 and κ_2 be the principal curvatures at vertex v, then: $\kappa_1 \kappa_2 = K$ and $\kappa_1 + \kappa_2 = 2H$. Therefore κ_1 and κ_2 are the roots of the polynomial $X^2 - 2H \cdot K + K$. The absolute curvature at v is defined by: $K_{\text{abs}} = |\kappa_1| + |\kappa_2|$. In our algorithm, this indicator is used throughout the vertex clustering process. Figure 2.1 presents absolute curvature fields for both a geological surface and the well known rocker arm model.
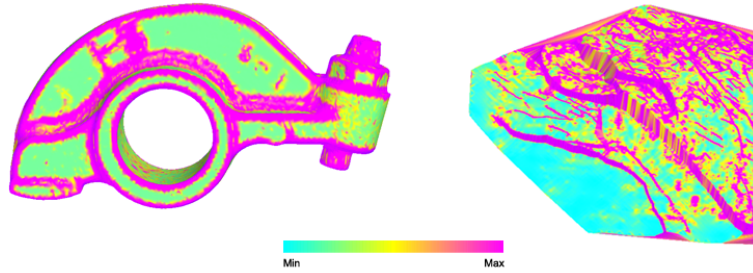

Figure 2.1: Discrete absolute curvature fields: left, a rocker arm - right, a geological surface

2.3.2. *Adaptive segmentation.* The spatial vertex partition is technically handled using a forest of BSP trees in order to control efficiently the size of the resulting mesh. Provided that each leaf of the BSP trees eventually produces a vertex, the leaves of the BSP tree are subdivided until the desired number of vertices is reached.

This process consists of three steps: initialization, adaptive segmentation, and last post-processing. Let us now detail each of them.

Initialization. After loading the mesh, the initialization step consists both in regularly segmenting the surface (subdividing the whole mesh by a 3D regular grid) and in computing for each vertex the corresponding absolute curvature indicator. The number of trees created corresponds to the number of cells of the uniform grid used for segmentation. Each root of this forest maintains a list of vertices and an absolute curvature value (defined as the sum of the absolute curvatures at the vertices of the cell).

Note that the size of the uniform grid does not directly control those of the resulting segmented mesh: this control arises from the adaptive segmentation step.

When the input data are voluminous, it is important that the size of the regular grid cells be small enough to simplify and accelerate the adaptive segmentation step. Moreover, in the sparse areas, the initial uniform clustering step prevents that too distant vertices be grouped by adaptive segmentation (which would result in distortions).

Adaptive segmentation of the mesh. Once the surface has been segmented by means of a regular grid (as described previously) we obtain an array of n BSP trees (where n is the number of cells of the initial regular grid). Moreover, these trees are sorted in a priority queue ordered by decreasing absolute curvature value.

The BSP tree is then iteratively updated as follows (let n be the number of leaves of the forest and let m be the number of vertices required for the simplified mesh):

While $n < m$:

    (1) *chose the leaf of maximal absolute curvature*
    (2) *create a subdivision plane by PCA analysis*
    (3) *subdivide the leaf according to this plane and update the BSP tree*

In order to determine a **subdivision plane** appropriate to the repartition of vertices in the cell (see [11]), we use a principal component analysis of the normals of the cell (see [13]).

Let us recall the main results on principal component analysis. Let $\{x_1, \ldots, x_n\}$ be a set of vertices. The covariance matrix of this set is defined by:

$$Z = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^\top$$

where $\bar{v}$ denotes the average of the set $\{x_1, \ldots, x_n\}$.

The eigenvectors of this matrix give the main variation directions of the set of vectors (for a cloud of points inscribed in a rugby ball, these directions are the axes of the ball). The eigenvector associated to the largest (resp. smallest) eigenvalue corresponds to the direction in which vectors spread out[1] the most (resp. the least).

In our setting, at each step of the adaptive process, the strongly bent cells are split in order to decrease their curvature as much as possible. Ideally, the subdivision plane should be orthogonal to the direction of maximal curvature (see figure 2.2). However, contrarily to figure 2.2, we are not interested in smooth surfaces but in cells issued from a triangular mesh. Therefore, it is necessary to find a discrete approximation of principal directions.
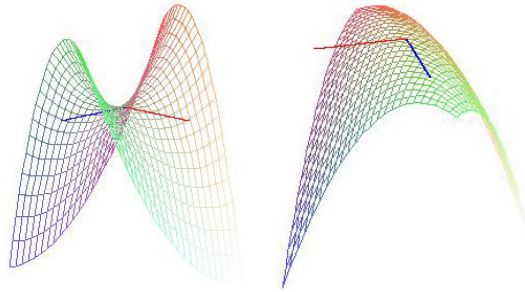

Figure 2.2: Principal curvatures on smooth surfaces: (in red, direction of maximal curvature - in blue, direction of minimal curvature)

Normal curvature in direction τ is the normal component of acceleration in this direction. Therefore, principal directions correspond to directions (in the tangent plane) of minimal and maximal variation of the normal vector.

In the discrete case, principal component analysis of the set of normals of the cell provides the main spreading directions of this set. Let e_1, e_2 and e_3 be unitary eigenvectors of the covariance matrix, associated to eigenvalues $\lambda_1 < \lambda_2 < \lambda_3$ (eigenvalues and eigenvectors are computed with the Jacobi method [25]) . Direction e_1 is that of minimal variance, therefore it approximates the average normal vector of the cell. Direction e_3 (orthogonal to e_1) is that of maximal variance. Thus it approaches the principal direction of maximal curvature and we will take e_3 **to be the normal of the splitting plane**.

Moreover, the affine subdivision plane should be inserted around the vertex of maximal curvature; but in order to split the cell efficiently, this vertex should not be too close from the border. Therefore, we insert the splitting plane at the **barycenter of the vertices weighted by their absolute curvature**. The resulting clustering is quite satisfactory both for large and small cells (see figure 2.3).

Once the subdivision plane is determined, the leaf corresponding to the considered cell in the BSP tree is split into two new leaves. Vertices of the original cell are assigned to one of these leaves depending on their position with respect to the splitting plane. Then, we assign each triangle to the set of cells its vertices belong (thus, a triangle generally belongs to up to three cells). The discrete surfaces we are studying are topologically connected. However, nodes can contain distinct disconnected components. In such a case, replacing the vertices of the cell by a single vertex would produce a non-manifold mesh; thus we test the connectivity of nodes and eventually split the non connected leaves into their connected components.

---

[1] The direction in which vectors spread out the most is actually the direction of maximal variance
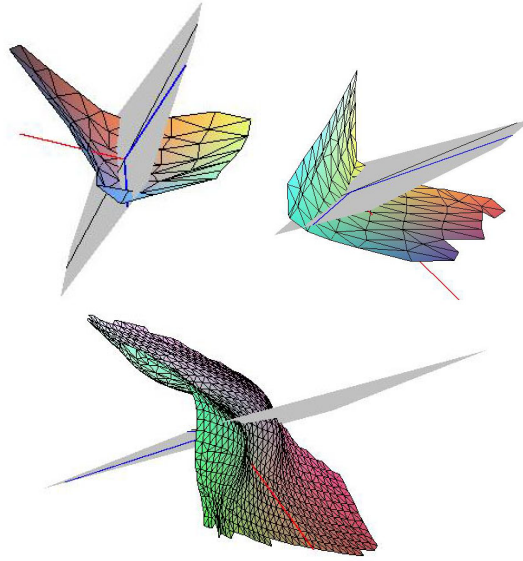
Figure 2.3: Splitting planes for small cells (top) and a 300 vertices cell

The following test is applied to each leaf of the BSP tree; the algorithm uses a list L (initially containing all the vertices of the leaf) and a queue f (initially empty).

- Get the head of L into v
- Insert v into f
- While f is not empty :
  - Get the head of f into v
  - For any v' neighbor of v:
    if v' belongs to L then
      * Insert v' into f
      * Remove v' from L

At the end of this test, if L is empty, the cell contains a single connected component and thus, the simplification process goes on normally. If L is not empty, the cell contains disconnected components. The leaf is split into two new leaves respectively containing the vertices still present in L and the others. The topological test goes on on the first set until all the connected components have been identified.

In spite of its cost, this test is necessary to guarantee the topological properties of the simplified surface. Figure 2.4 presents both the uniform cells and those obtained after the adaptive subdivision process.
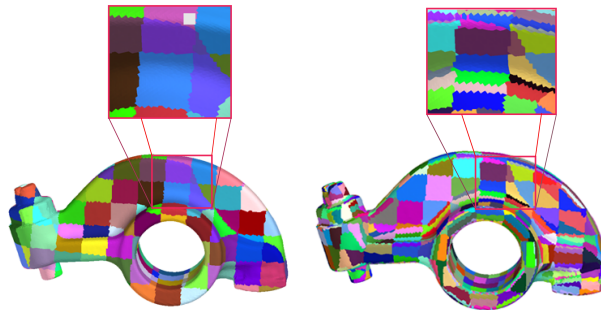
Figure 2.4: Results of the adaptive subdivision process: left, uniform clustering - right, adaptive clustering

Post-processing. Once the cells have been split and the expected decimation rate is reached, a representative vertex must be computed for each of them (together with an appropriate topology, inherited from the original mesh).

In order to approximate cells as precisely as possible, we use a method similar to [7], [17] and [22]. For each cell, we define a quadratic form (called *quadratic error metric*) estimating the distance between any point of space and the cell. The optimal position of the representative vertex is obtained by minimization of this quadratic form.

Let us now define this quadratic form. For any triangle t in the cell, let $\mathcal{P}_t$ be the plane defined by t, the quadratic form $Q_t : \mathbb{R}^3 \to \mathbb{R}$ associated to t is defined by $Q_t(v) = \mathrm{d}(v, \mathcal{P}_t)^2$. The cartesian equation of $\mathcal{P}_t$ can be written: $n^\top v + d = 0$ where n denotes the unitary normal of t and d is a constant. The distance $d(v, \mathcal{P}_t)^2$ can thus be written as $d(v, \mathcal{P}_t)^2 = v^\top (nn^\top)v + 2(dn^\top)v + d^2$. Let us define:

$$Q_t(v) = v^\top A_t v + 2B_t^\top v + C$$

with $A_t = nn^\top$, $B_t = dn^\top$ and $C_t = d^2$.

The quadratic form associated to a cell is the sum of the forms associated to each of its triangles. As a consequence, it can also be written: $\mathcal{Q}(v) = v^\top Av + 2B^\top v + C$. Figure 2.5 presents quadratic error metrics for different cells. The red axes represent the axes of Q; they originate at the point $v_{\min}$ minimizing Q (let $\varepsilon_{\min} = Q(v_{\min})$). The isosurface $Q = 1.5 \times \varepsilon_{\min}$ is represented in black.

Observe that the axes produced by the principal component analysis of the cell (represented in blue) are quite similar to the axes of the quadratic error metric[2].
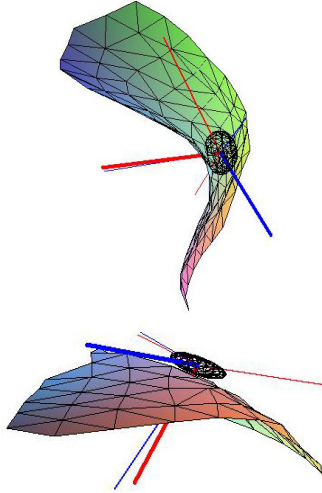
Figure 2.5: Quadratic error metric for different cells - top: a saddle cell - bottom: a convex cell

We have $\mathrm{d}Q(v).h = 0$ and as matrix A is symmetric and non negative, minimizing Q comes to solving $Av + B = 0$. This linear system is solved by singular values decomposition: $A = U\Sigma V^\top$ where Σ is a diagonal matrix and U and V are orthogonal matrices. Let us define matrix Σ^+ by:

$$(\Sigma^+)_{i,j} = \begin{cases} \frac{1}{\Sigma_{i,j}} & \text{if } \Sigma_{i,j} \neq 0 \\ 0 & \text{else} \end{cases}$$

Let $\hat{x}$ be the barycenter of the cell. The closest point to $\hat{x}$ satisfying equation $Ax + B = 0$ is given by:

$$x = \hat{x} - V\Sigma^+ U^\top (B + A\hat{x})$$

---

[2]Which is not so surprising as

$$A = \sum_{t \in \text{cell}} n_t n_t^\top \text{ whereas } Z = \frac{1}{k-1} \sum_{t \in \text{cell}} (n_t - \bar{n})(n_t - \bar{n})^\top$$

where n_t denotes the normal of triangle t and $\bar{n}$ the average normal of the cell.

Once this representative vertex is determined for each cell, it remains to rebuild a topology over these vertices, inherited from the initial topology of the surface. The algorithm is as follows:

For any face f in the initial mesh:

- *if f belongs to three different cells, it is kept,*
- *otherwise, it is degenerate (reduced to a segment or vertex in the new mesh) and therefore, it is removed.*

The remaining faces generate the topology over the set of representative vertices and the quadratic error metric of each cell becomes that of its representative vertex.

Let us point out that this post-processing (also used by [7], [17] and [22]) does not guarantee the manifoldness of the result (only that generally, it is manifold). The following example (figure 2.6) illustrates such a topological problem. The initial mesh (drawn in black on the left figure) is split into four cells and thus, the simplified mesh (in red) is not a manifold. Flipping edge (e, i) solves the problem (see right figure).
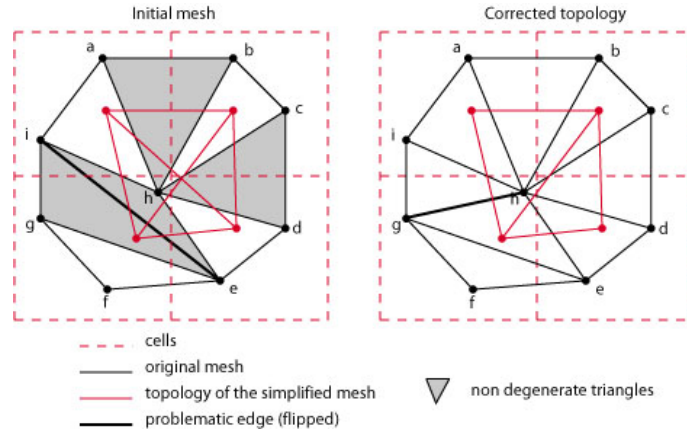
Figure 2.6: Heuristic for the well known topological problem (non-manifoldness): left, the original mesh - right, the corrected mesh (an edge has been flipped) which gives rise to a manifold simplified mesh

Our idea is to detect and avoid edges causing non-manifoldness, and actually, edges of the original mesh belonging to two triangles that will be non degenerate are one of the main cause for such problems (as they produce crossing edges). Therefore, before building the topology of the simplified mesh, we apply the following heuristic to the initial mesh:

(1) *select the edges (v_1, v_2) of the initial mesh incident to two different non degenerate triangles $((v_1, v_2, v_3)$ and $(v_1, v_2, v_4))$; these edges are responsible for non-manifoldness*

(2) *for each of these edges:*
  if (v_3, v_4) belongs to a single cell
    flip (v_1, v_2) ((v_1, v_2) is replaced by (v_3, v_4)):

In the previous example, only edge (e, i) is concerned and its flip makes the simplified mesh a manifold surface.

All this data (representative vertices, topology and quadratic error metric) is transmitted to the second step of our simplification algorithm.

2.4. **Iterative edge collapse.** The second step of our algorithm consists in simplifying more finely (by iterative edge collapse) the intermediate mesh previously obtained. We apply the method introduced by Garland and al. ([10]) with the quadratic error metrics previously computed.

Contracting a pair of vertices $(v_1, v_2) \rightarrow \bar{v}$ consists in replacing the vertices v_1 and v_2 by a new vertex $\bar{v}$ minimizing the resulting error (where error is measured with the quadratic error metric just described). Vertex $\bar{v}$ is then linked with the neighbors of v_1 and v_2.

Let us now come into details. The quadratic error made on the edge (v_1, v_2) is estimated by $Q_{(v_1, v_2)}(v) = Q_{v_1}(v) + Q_{v_2}(v)$. The algorithm is as follows:

- *For any edge (v_1, v_2), compute $\bar{v}$ the vertex minimizing error $Q_{(v_1,v_2)}(v)$. The cost of contraction $(v_1, v_2) \to \bar{v}$ is defined as $Q_{(v_1,v_2)}(\bar{v})$.*
- *Order the pairs in a stack by increasing order.*
- *While the desired decimation rate is not reached:*
  - *remove the pair (v_1, v_2) of lower cost from the stack,*
  - *contract this pair; the quadratic error metric associated to the new vertex $\bar{v}$ is*
    $$Q_{\bar{v}} = Q_{v_1} + Q_{v_2}$$
  - *update the contractions (position of the optimal vertices) and their costs for the 1-neighbor ring of $\bar{v}$*

2.5. **Results.** The performances of the simplification process strongly depend on the following parameters: first the size of the intermediate mesh (that is the simplified mesh obtained after the first step), second, the size of the uniform grid.

The size of the uniform grid must not be too small, otherwise, the following adaptive subdivision makes no more sense and wouldn't improve uniform segmentation anymore. However, this parameter provides a control over the errors made by adaptive segmentation: at worst, after the adaptive segmentation step, the size of the cells equals those of the grid. In practice, a good choice for the size of the cells is to take them between 1.5 and 2 times the average length of the edges. As for the size of the intermediate mesh, we experimentally choose a ratio between 0.5 and 0.8 of the size of the initial mesh. Both parameters must actually be chosen in order to let enough "place" to both steps to work over the data.

As one can observe (figure 2.7 and 2.8, the simplified surfaces are visually very satisfactory; actually, they are very close to those obtained by a pure iterative edge contraction - this will illustrated when studying the Hausdorff distance between the initial surface and the simplified one. Observe that the sharp edges are well preserved. For geological surfaces, it is essential as
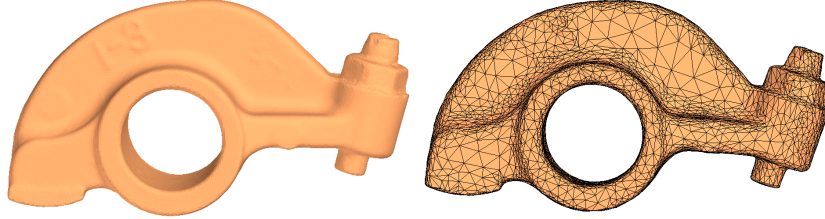
Figure 2.7: The rocker arm model simplified by our method: initial model, 40k vertices (left) - simplified model, 5k vertices, $D_{\max} = 0.00029$, $D_{\text{avg}} = 0.0000345$ (right) - size of the uniform grid: 41x24x78, size of the intermediate mesh: 20088 vertices

these characteristic lines are of particular interest for the geological interpretation of surfaces.
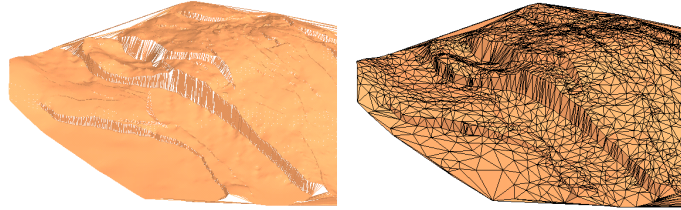
Figure 2.8: A geological surface simplified with our method: initial model, 112k vertices (left) - simplified model, 3k vertices - size of the uniform grid: 151x188x27, size of the intermediate mesh: 56136 vertices

In order to estimate the quality of our results, we have first compared them with those obtained by Shaffer and Garland with their mixed approach ([11]). The tests have been performed with two models: the "lucky lady" model (500k vertices) and the "dragon" model (437k vertices). Table 2.9 presents the numerical results obtained for this comparison.

Model	V_{in}	V_{out}	(1)/(2)	Grid size	Occupied cells	1st phase mesh size	Non-manifold edges	Time(s)	Error	Gain (2)/(1)
Venus	134k	20k	(1)	78x104x93	31464	31464	210	13	$D_{avg} = 0.17E-3$ $D_{max} = 4.75E-3$	
Fig. 9			(2)	62x84x74	20975	31507	84	14	$D_{avg} = 0.17E-3$ $D_{max} = 2.37E-3$	$G_{avg} = 2,13\%$ $G_{max} = 50,12\%$
Venus	134k	3k	(1)	62x84x74	20975	20975	190	12	$D_{avg} = 0.39E-3$ $D_{max} = 4.94E-3$	
			(2)	52x70x62	15072	20975	114	16	$D_{avg} = 0.38E-3$ $D_{max} = 2.80E-3$	$G_{avg} = 2,92\%$ $G_{max} = 43,46\%$
Lucky	500k	45k	(1)	182x311x105	119713	119713	5457	83	$D_{avg} = 0,21$ $D_{max} = 5,35$	
			(2)	145x249x84	79190	119700	1587	86	$D_{avg} = 0,19$ $D_{max} = 2,04$	$G_{avg} = 8,24\%$ $G_{max} = 61,93\%$
Lucky	500k	100k	(1)	242x414x139	197251	197251	4579	82	$D_{avg} = 0,12$ $D_{max} = 5,33$	
Fig. 10			(2)	182x311x105	119713	194725	866	96	$D_{avg} = 0,11$ $D_{max} = 1,16$	$G_{avg} = 7,20\%$ $G_{max} = 78,28\%$

(1) Garland and al. - (2) our method

- V_{in} is the size of the initial model and V_{out} is that of the simplified model
- *Grid size* is the size of the uniform grid
- *Occupied cells* is the number of leaves in the final BSP tree
- 1st *phase mesh size* is the size of the intermediate mesh
- *Time (s)* is the running time of the whole simplification process
- *Error*: D_{max} is the Hausdorff distance between the original and the simplified mesh - D_{avg} is the average symmetric distance between both models (see[3] for more details)
- *Non-manifold edges* is the number of non-manifold edges resulting from vertex clustering (our heuristic aims at avoiding them)

Figure 2.9: Numerical comparison between our algorithm and Garland and al. 2002.

Figure 2.11 and 2.10 present the related graphical results. Observe that besides the numerical results, our method visually preserves well the sharp folds of the models and produces regular meshes.
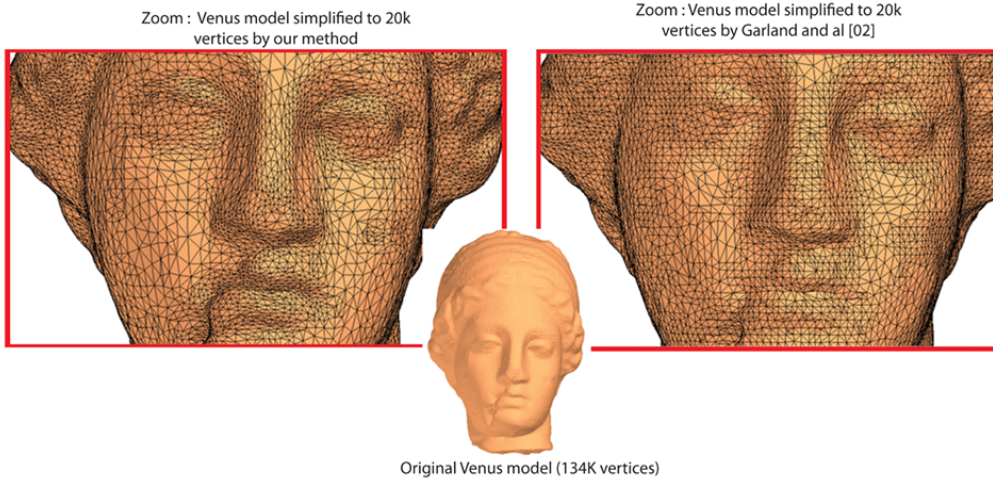
Figure 2.10: Comparison of our method and Garland and al. 2002 - the "venus" model (134k vertices) - simplified model: 20K vertices

In order to estimate the quality of our simplified meshes, we have compared them with surfaces obtained by the pure iterative edge collapse algorithm ([10]). Figure 2.12 presents running times and error maps for both of these algorithms.

Therefore, the quality of our results is similar to [10] whereas our running time is three times lower.

3. FEATURE LINE EXTRACTION

3.1. **Context.** The skeleton is a robust shape descriptor faithfully characterizing the topology and the geometry of an object. This notion is widely used for various applications such as video tracking [9], shape recognition [27], surface sketching [18], and in many other scientific domains. Several techniques have been proposed to extract the skeleton from binary 2D images [28], 3D
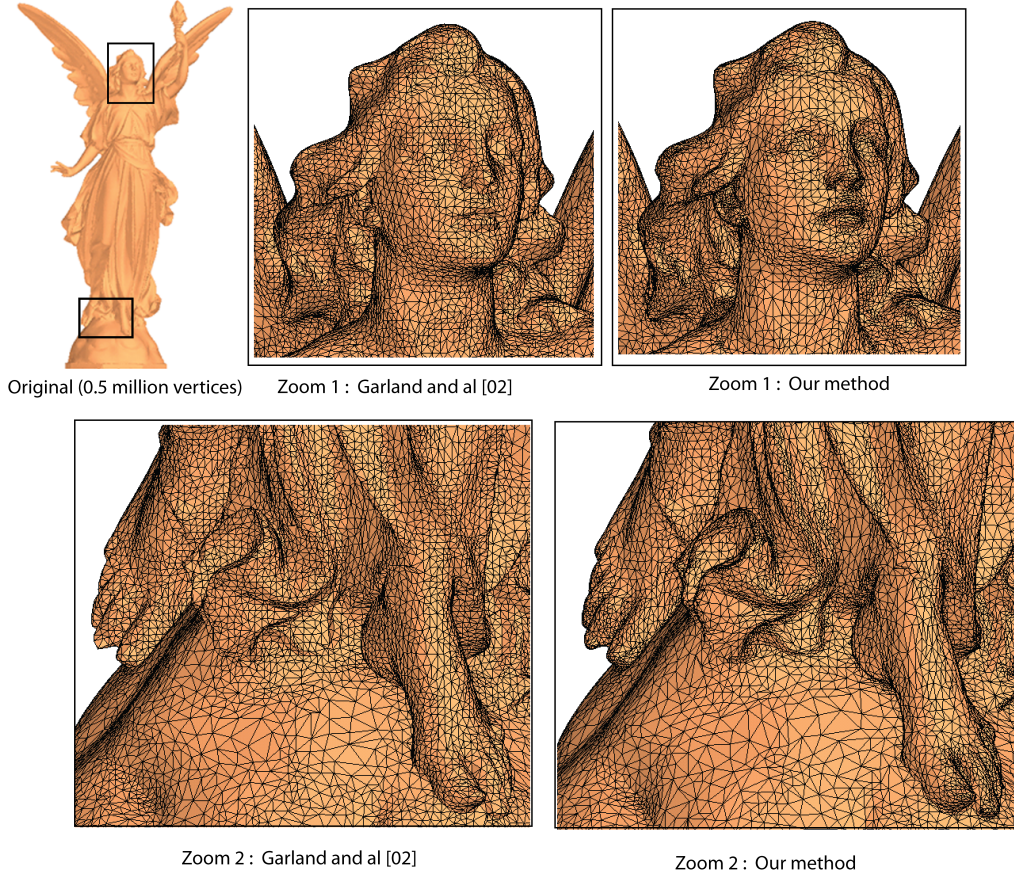
Figure 2.11: Comparison of our method and Garland and al. 2002 - the "lucky lady" model (500k vertices) - simplified model: 100K vertices
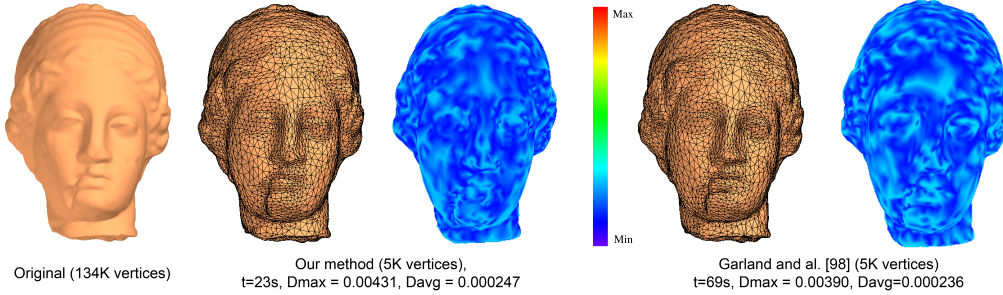
Figure 2.12: Comparison of the map of errors for our method and Garland and al. 1999 - the "venus" model (134k vertices)

closed meshes defining a volume [3], or 3D cubic grids [16]. However few have been dedicated to the extraction of skeletons from a binary information located on an arbitrary triangulated mesh. Rössl *et al.* [21] have presented a method in which some mathematical morphology operators have been ported to triangulated meshes. The main interest of this approach is to combine an efficient computation and a simple implementation. However, regarding the operator definitions and the underlying algorithm, several drawbacks have been pointed out which mainly lead to unexpectedly disconnected skeletons [15].

In this work, we propose a novel method to extract the skeleton of unstructured mesh patches

by a topological thinning process. To figure out the issues of skeletonization of heterogeneous and arbitrary triangulated meshes, we extend the concepts introduced in [21]. The presented approach herein strictly relies on the mesh connectivity to achieve the extraction of the final skeleton. Therefore, for the sake of understanding, the basic method of Rössl *et al.* is described in Section 3.2 with an assessment of its abilities and drawbacks. Section 3.3 details the proposed approach and introduces the additional definitions and the novel algorithm. The results of our method including tests on irregular meshes as well as on the performance of the algorithm are shown in Section 3.4. Finally, an application to feature line detection is presented in Section 3.5.

3.2. **Basic notions and definitions.**

3.2.1. *Position of the problem.* Let $\mathcal{S}$ be an arbitrary manifold surface represented by an unstructured mesh patch $\mathcal{M}$ such as $\mathcal{M} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$. The sets $\mathcal{V}, \mathcal{E}$, and $\mathcal{T}$ correspond, respectively, to the vertices, the edges, and the triangles composing $\mathcal{M}$, the piecewise linear approximation of $\mathcal{S}$. The vertices are denoted by p_i, with $i \in [0; n[$ and $n = |\mathcal{V}|$ being the total number of vertices of $\mathcal{M}$. The neighborhood $\mathcal{N}$ of a vertex p_i is then defined as following:

$$(3.1) \qquad \mathcal{N}(p_i) = \{q_j \mid \exists \text{ a pair } (p_i, q_j) \text{ or } (q_j, p_i) \in \mathcal{E}\}.$$

In such a case, $m_i = |\mathcal{N}(p_i)|$ represents the total number of neighbors of p_i.

Let now consider a binary attribute F on each vertex of $\mathcal{V}$. The set $R \subseteq \mathcal{V}$ is then written as follows:

$$(3.2) \qquad \forall p_i \in R \iff F(p_i) = 1.$$

The attribute F may be defined from beforehand process such as a manual selection, or a thresholding based on geometrical properties (triangle area, principal curvatures, *etc.*). Then, an edge $e = (p, q)$ belongs to R if and only if $p, q \in R$. Similarly, a triangle $t = (p, q, r)$ belongs to R if and only if $p, q, r \in R$.

The main objective is to finally develop a technique to extract the skeleton of the set R by using a topological thinning based on the mesh connectivity.

3.2.2. *The existing approach.* The skeletonization algorithm introduced by Rössl *et al.* consists in an iterative constraint thinning. This relies on a classification of each vertex of R. The authors proposed then three vertex types and $c(p_i)$, the *complexity* of the vertex p_i such as:

$$(3.3) \qquad c(p_i) = \sum_{j=0}^{m_i - 1} |F(q_j) - F(q_k)|,$$

where $k = j + 1 \bmod m_i$ and $q_j, q_k \in \mathcal{N}(p_i)$.

Definition 1. A vertex p_i is considered as *complex* if and only if $c(p_i) \geq 4$. The set of all *complex* vertices is named C.

A *complex* vertex p_i thus potentially corresponds to a part of a skeleton branch if $c(p_i) = 4$, or a connection through several branches if $c(p_i) > 4$.

Definition 2. A vertex p_i is marked as *center* if and only if and $\mathcal{N}(p_i) \subseteq R$. The set of all *center* vertices is named E.

Definition 3. A vertex p_i is called *disk* if and only if $\exists q_j \in \mathcal{N}(p_i), q_j \in E$ that is a *center*. The set of all *disk* vertices is named D.

A *disk* vertex corresponds to a *simple* point: a point that does not modify the expected skeleton topology if it is removed [6]. We denote $\overline{X}$ the complementary of the set X in the region R.

Definition 4. The *skeletonization operator* of R is defined as a constrained thinning:

$$(3.4) \qquad skeletonize(R) = R \setminus (D \cap \overline{C \cup E}).$$

After applying the skeletonization operator until idempotence on R, the set of the remaining vertices, corresponding to the final *skeleton*, is called Sk_R. During each pass, the skeletonization operator removes the boundary *disk* vertices. Figure 3.1 illustrates the execution of the algorithm. After obtaining the skeleton Sk_R of R, it is possible to remove the smallest branches. This last operation is called *pruning* and defined as follows:

$$(3.5) \qquad\qquad\qquad prune(Sk_R) = Sk_R \setminus \overline{C}.$$
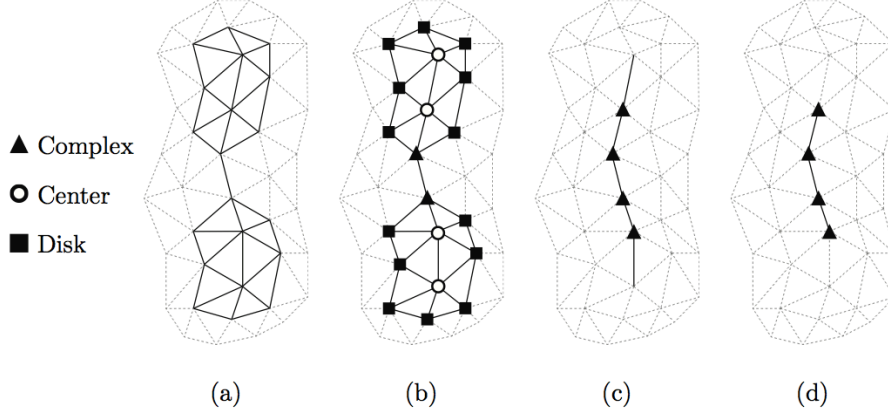
This pruning step is shown by Figure 3.1 (d).

Figure 3.1: Illustration of the Rössl *et al.* algorithm. From left to right: (a) a set of vertices R, (b) classification of R, (c) thinning until idempotence, and (d) resulting skeleton after pruning.

3.2.3. *Result assessment.* Due to the simplicity of the used operators, the computational time of the Rössl *et al.* method is very low, and the skeleton extraction is thus almost instantaneous on meshes composed of 50K triangles. However, the accuracy and the continuity of the obtained skeleton deeply depends on the mesh configuration. In other words, a same set R defined on two different triangulations of S could lead to skeletons with two topologies drastically different. Moreover, the lack of continuity also occurs in the case of particular configurations that are shown in Figure 3.2 because the removal of *disk* vertices can modify the topology of the skeleton. Figure 3.3 illustrates the unexpected results and disconnections generated by the execution of the skeletonization. Once the vertices P_1 and P_2 are removed (b), the skeleton becomes disconnected at this location (c). However, some vertices would change to *complex* if a new classification step was applied. This kind of vertices represents relevant points in a topological point of view and thus, should not be deleted.

Another issue occurs since pruning is applied: the ending vertices of the skeleton are removed. As a matter of fact, when the set R contains no *center* and no *complex* vertex, the pruning operator removes all the vertices. This case is illustrated by Figure 3.4.

3.3. **A skeletonization method for any arbitrary triangulated mesh.** Both a new definition of particular vertices and a new algorithm have been elaborated to solve the disconnection issues previously raised up in Section 3.2. These two key points of the approach we propose are successively presented below.

3.3.1. *Additional definitions.* The different classes of vertices proposed by Rössl *et al.* aim at describing the topology of R. However, they are not sufficient as there are still vertices that are unmarked and that are then not considered in the skeletonization. For this reason, we introduce the *outer* class.

Definition 5. A vertex p_i is marked as *outer* if and only if $F(p_i) = 1$ and $p_i \notin (C \cup D \cup E)$. The set of *outer* vertices is named O and is defined as follows:

$$(3.6) \qquad\qquad\qquad O = R \setminus (C \cup D \cup E)$$
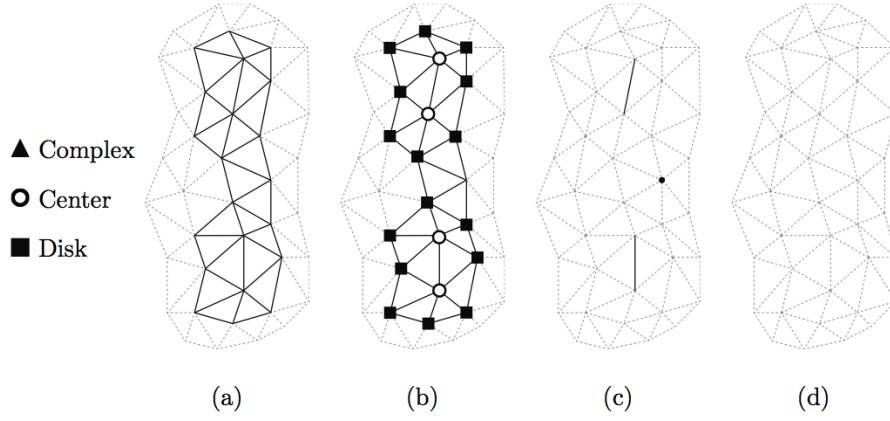
Figure 3.2: Example of unexpected results by applying the Rössl *et al.* method. From left to right: (a) the set of feature points R, (b) classification of R, (c) skeletonization of R, (d) resulting skeleton after pruning.
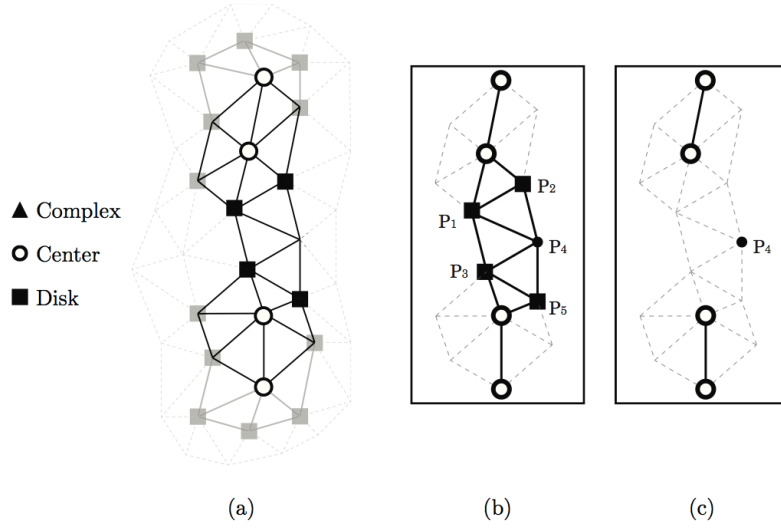
Figure 3.3: Execution of the skeletonization operator [21]: (a) vertex classification, (b) execution of the algorithm, (c) final skeleton with a broken topology.
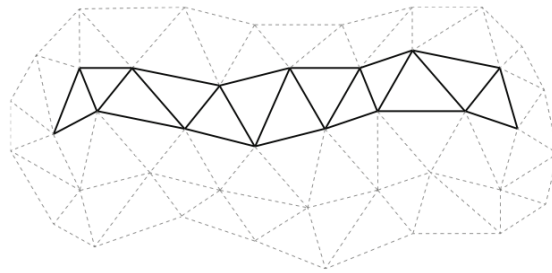
Figure 3.4: Example of a particular configuration: while the vertices of R are not classified, they will be deleted by the pruning operator of Rössl *et al.*

As it has been shown previously, a vertex may change from one class to another and, as a side-effect, this may lead to potential disconnections during the skeletonization. To counteract this issue, we propose to define a priority between the classes.

Definition 6. The *disk* class has a lower priority over the other classes.

If a vertex is already classified as *disk*, it can change to *complex*, *center* or *outer* if necessary.

3.3.2. *Algorithm.* If the skeletonization operator defined by Rössl *et al.* is directly applied to an unstructured patch, the final result may suffer from disconnections as some *disk* vertices are deleted while they characterize the topology of the object. To correct this issue, the algorithm we propose does not remove *all* the *disk* vertices but only those that will not be converted to a different priority class after the operator application. This requires to add an additional step in the algorithm: at each application of the skeletonization operator, the class of a vertex is recomputed before its deletion. For example, if a *disk* vertex becomes a *complex* vertex, the vertex is not removed.

However, the resulting skeleton may be too thick using this technique (*e.g.* if it is composed of only *outer* vertices). For this reason, a final cleaning step is added to obtain the expected skeleton. At this stage, the skeleton must be composed of *complex* vertices (*i.e.* the skeleton branches or nodes) and *outer* vertices, the ending points of the branches with only one *complex* vertex in their neighborhood. Thus, to obtain the final skeleton, a two steps process is applied:

- the *outer* vertices that have more than two neighbors belonging to R are removed;
- the *outer* vertices with at most one neighbor belonging to R are kept.

Moreover, as for the skeletonization operator, each vertex complexity change is checked before removing this vertex. Examples of resulting skeletons are shown in Figure 3.6 and the impact of the algorithm modification with the update step is presented in Figure 3.7: *disk* vertices are deleted (b) after checking their classes (c). During the deletion of P_1 and the update step, the class of P_2 changes from *disk* to *complex* and P_4 from *outer* to *complex*. Thus, these vertices are not removed and the extracted skeleton is fully connected and faithfully characterizes the topology of R (d). The complete method of skeleton extraction is summarized by the algorithm presented on Figure 3.5.

3.4. **Results.** Some results of skeleton extraction on meshes are presented in Figures 3.8, 3.9 and 3.10. The obtained skeletons describe the geometry and the topology of the original set R. The used meshes are relatively homogeneous in Figure 3.8 while, in Figures 3.9 and 3.10, the algorithm has been tested on irregular meshes to show the robustness of the proposed approach to unstructured meshes. It may be noticed that the resulting skeletons are the expected ones and reflect correctly the topology and geometry of the original set R in a proper way.

Moreover, since the definitions and the operators used to extract the skeleton are very simple, the computational time of the proposed approach is also very low, even if an additional checking step has been added. It is possible to process a mesh with 100K vertices in 1 second. The tests have been ran on an Intel Core 2 Duo 2.8 Ghz.

To complete the algorithm tests and to evaluate the robustness of the proposed approach, an application dedicated to the feature line detection is presented in the following section.

3.5. **Application to feature line detection.** The detection of features within 3D models is a crucial step in shape analysis. It is possible to extract from the surface of an object simple shape descriptors such as lines (drawn on the surface). Generally, the methods of feature line detection focus on the estimation of differential quantities and the research of curvature extrema. However, these techniques are based on *third-order* differential properties and it leads to a common issue: they produce disconnected feature lines because of flat and spherical areas and because of the noise present in data sets. Thus, it is particularly difficult to generate intersections between feature lines. To overcome these recurrent issues, we propose to apply our method to extract salient lines of a model.

In order to define sets over triangulated 3D meshes, we use the algorithm proposed by Kudelski *et al.* [14]. We compute the mean curvature H through a local polynomial fitting in the least-squares sense [12]. The binary attribute F is then defined at each vertex p_i as follows:

$$(3.7) \qquad\qquad H_{p_i} > 0 \implies F(p_i) = 1.$$

repeat

   forall the *vertices* $p_i \in R$ **do**

      if p_i *is a* disk *vertex* **then**

         compute the complexity $c(p_i)$ of the vertex

         if *the priority of* p_i *does not change* **then**

            delete p_i

until *idempotence*

repeat

   forall the *vertices* $p_i \in R$ **do**

      if p_i *is an* outer *vertex* **then**

         compute the complexity $c(p_i)$ of the vertex

         if *the priority of* p_i *does not change and if* $|\mathcal{N}(p_i)| > 2$ **then**

            delete p_i

until *idempotence*

repeat

   forall the *vertices* $p_i \in R$ **do**

      if p_i *is an* outer *vertex* **then**

         compute the complexity $c(p_i)$ of the vertex

         if *the priority of* p_i *does not change and if* $|\mathcal{N}(p_i)| > 1$ **then**

            delete p_i

until *idempotence*

Figure 3.5: Extraction of the skeleton.

▲ Complex

◯ Center

■ Disk

★ Outer

       (a)   (b)   (c)   (d)
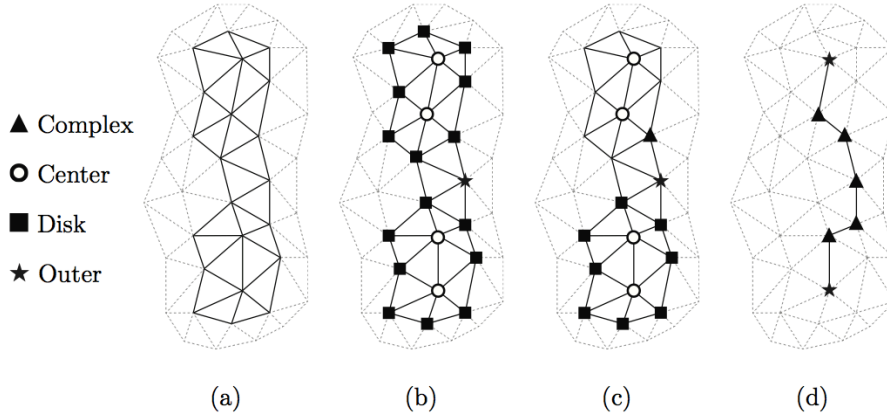
Figure 3.6: Illustration of the proposed approach: (a) region R, (b) vertex classification, (c) execution of the thinning algorithm with update, (d) final skeleton fully connected.

Finally, the objective is to thin the set, corresponding to potential feature parts of the mesh, in order to obtain lines describing the geometry and the topology of the object.

Figure 3.11 illustrates the process of feature line detection. The obtained characteristic lines are fully connected and describe accurately the topology of the sets. Then, due to the use of second-order differential properties (*i.e.*, the mean curvatures), the feature extraction is more robust.
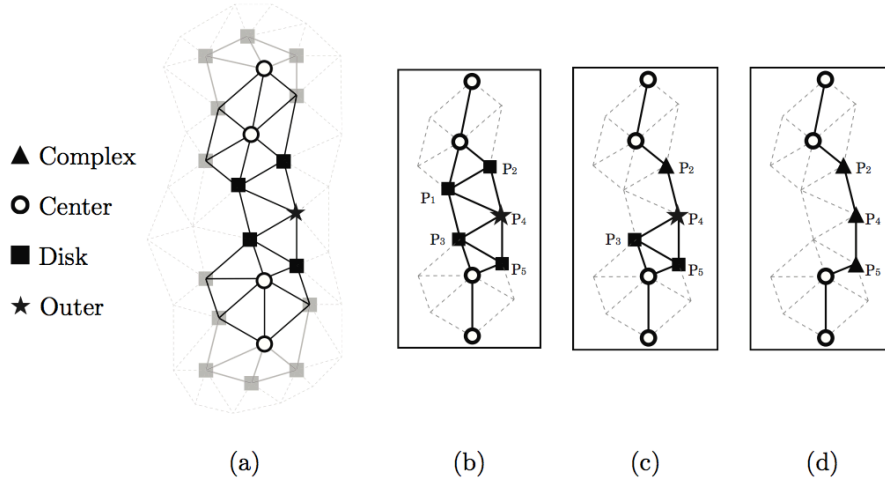
Figure 3.7: Detailed view of the thinning process: (a) vertex classification, (b) execution of the skeletonization operator, (c) update of vertex classes after deletion, (d) final skeleton.
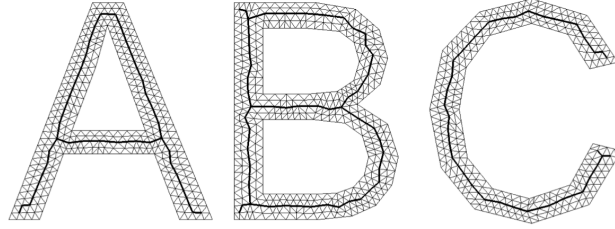
Figure 3.8: Application of the skeletonization algorithm on regular triangulated 3D meshes.
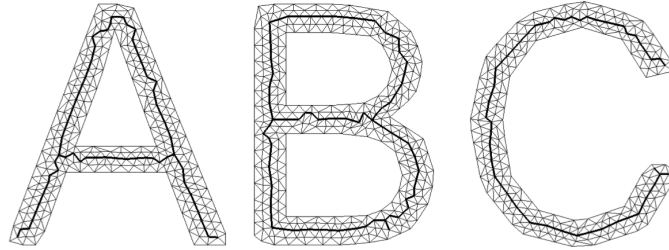
Figure 3.9: Skeleton extraction on irregular 3D meshes.

Moreover, this type of approach allows to generate intersections between feature lines, which is not possible with classical approaches (Figure 3.12).

4. Conclusion

We illustrated the discrete curvatures concepts with two applications. In the first one, our algorithm proposes an alternative to vertex clustering simplification methods and to iterative edge collapse methods, by a compromise between both approaches. Regarding the results presented
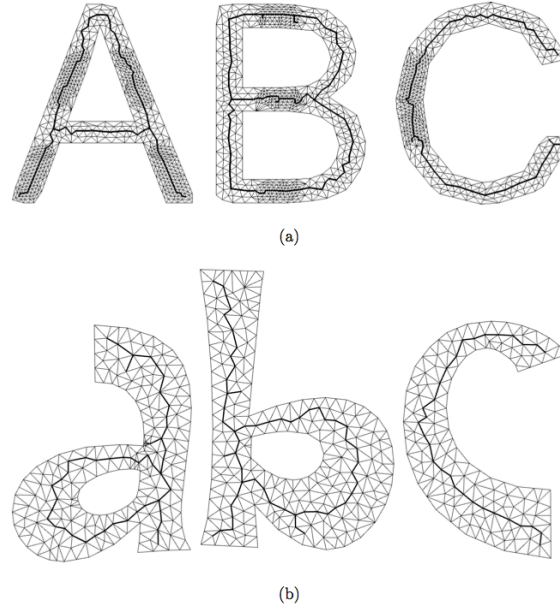
Figure 3.10: Extraction of the skeletons on meshes with mixed and unstructured meshes.

in section 2.5, this objective is reached. The main interest of this approach is to provide results of high quality (very similar to those obtained by an iterative edge collapse method) but with lower running times (by factors around 3 and up to 5) and memory consumption. Actually, our algorithms behaves well as for the average errors between the original and the simplified mesh and the maximal errors are significantly reduced compared to [11]. Moreover, the heuristic we apply in order to avoid the well known topological problems resulting from simplification based on vertex clustering proves quite efficient.

Our second application is an efficient and general new algorithm to extract the skeleton of a set R defined on a triangulated mesh by topological thinning. This approach relies on the definitions presented by Rössl *et al.* [21]. However, the latter generates, for some mesh configurations, unexpected skeletons that are generally more disconnected than they should. To overcome this issue, an additional definition of vertex categories has been added. Then, we have improved the thinning process by integrating a priority between vertex classes. Tests applied on different categories of meshes illustrate the efficiency of the approach. As future work, a formal proof based on [5] and issued from the notion of *simple vertices* (by analogy to *simple points*) may need to be considered. The Rössl *et al.* article does not include formal validations because the vertices classification is incomplete. With the changes made, the *disk* vertices truly correspond to simple points lying on a discrete 2-manifold. Thus it will be possible to transpose the notion of geodesic neighborhood to define topological numbers associated with simple vertices. A second prospect is related to the position of the skeleton nodes. Indeed, the defined operators do not integrate any geometrical information and the extraction of the skeleton only relies on a one-ring neighborhood. However, as the position of the skeleton is generally easier to correct than the topology, post-processing steps could be envisaged to optimize the skeleton position. In this way, the resulting skeleton will describe in a better way both the topology and the geometry of the set lying on the mesh.

Even if interesting and relevant results can already be obtained, we are aware than many theoretical works a practical experiments are still required to handle the open issues linked to large and noisy discrete objet analysis.
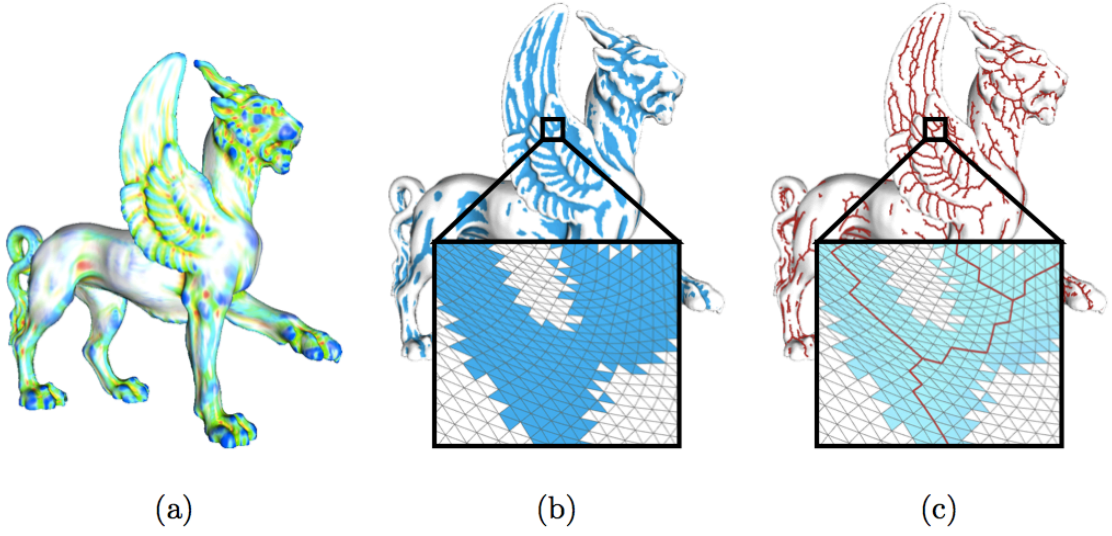
Figure 3.11: Algorithm of feature lines extraction: (a) curvature estimation, (b) definition of the set R, (c) extraction of lines from R by the proposed thinning approach.

(a) DeCarlo et al.

(b) Yoshizawa et al.

(c) Kudelski et al.

(d) Our approach.
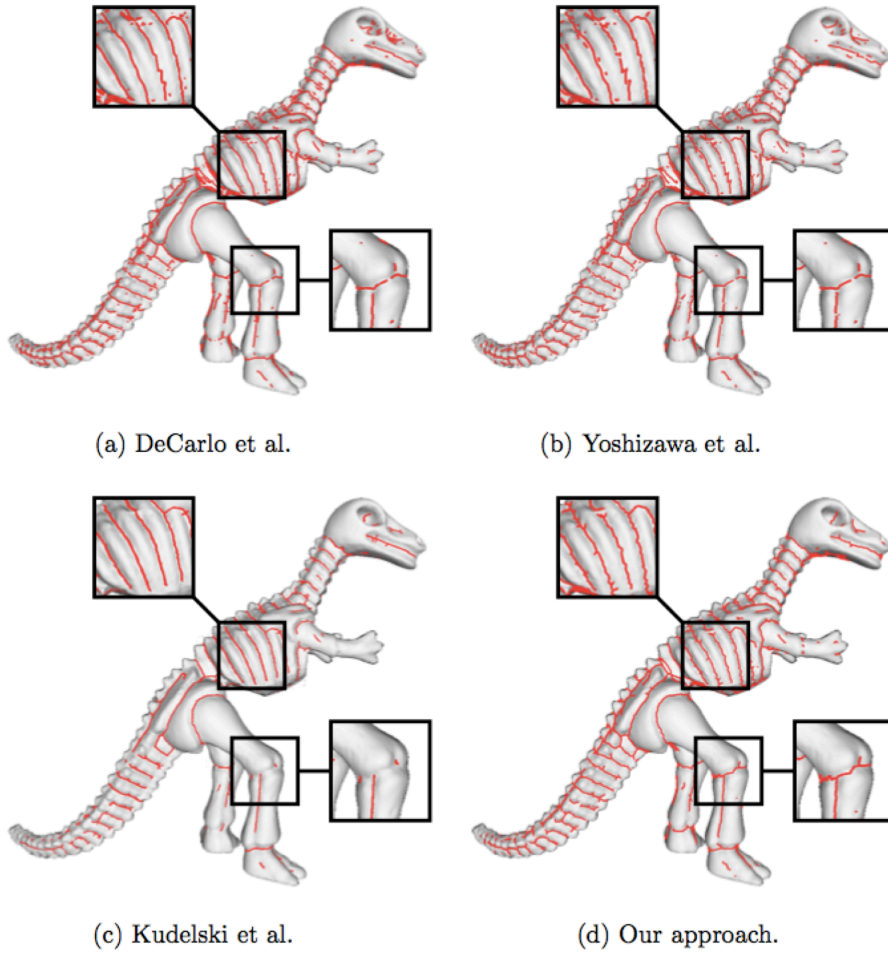
Figure 3.12: Comparison of results obtained from feature detection applied on *Dinosaur*.

References

[1] M. Daniel A. Bac and J.L. Maltret. 3D modeling and segmentation with discrete curvatures. *Medical Informatics and Technology*, pages 13–24, 2005.

[2] A. Alexandrov. Intrinsic geometry of surfaces. *Transactions of mathematical monograph AMS*, 1967.

[3] Oscar Kin-Chung Au, Chiew-Lan Tai, Hung-Kuo Chu, Daniel Cohen-Or, and Tong-Yee Lee. Skeleton extraction by mesh contraction. *ACM Transaction on Graphics*, 27(3):1–10, August 2008.

[4] A. Bac, N-V. Tran, M. Daniel, and J-F. Rainaud. Traitement de surfaces géologiques pour la construction de modèles 3D. In *journées du GTMG*, pages 22–23, Poitier, 03 2005.

[5] Gilles Bertrand. Simple points, topological numbers and geodesic neighborhoods in cubic grids. *Patterns Recognition Letters*, 15:pp. 1003–1011, 1994.

[6] Gilles Bertrand. A boolean characterization of three-dimensional simple points. *Pattern Recognition Letters*, 17:115–124, 1996.

[7] D. Brodsky and B. Watson. *Model Simplification In Reverse, Vector Quantization*. PhD thesis, University of Alberta, 2000.

[8] F. Cazals and M. Pouget. Estimating differential quantities using polynomial fitting of osculating jets. *Comput. Aided Geom. Des.*, 22(2):121–146, 2005.

[9] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'09)*, pages 1746–1753. IEEE Computer Society, June 2009.

[10] M. Garland. *Quadric-Based Polygonal Surface Simplification*. PhD thesis, Carnegie Mellon University, 1999.

[11] M. Garland and E. Shaffer. A multiphase approach to efficient surface simplification. In *VIS '02: Proceedings of the conference on Visualization '02*, pages 117–124, Washington, DC, USA, 2002. IEEE Computer Society.

[12] J. Goldfeather and V. Interrante. A novel cubic-order algorithm for approximating principal direction vectors. *ACM Transaction on Graphics*, 23(1):45–63, 2004.

[13] I. Jolliffe. *Principal component analysis*. Springer Verlag, 1986.

[14] Dimitri Kudelski, Jean-Luc Mari, and Sophie Viseur. 3D feature line detection based on vertex labeling and 2D skeletonization. In *IEEE International Conference on Shape Modeling and Applications (SMI'10)*, pages 246–250. IEEE Computer Society, June 2010.

[15] Dimitri Kudelski, Jean-Luc Mari, and Sophie Viseur. Extraction of feature lines with connectivity preservation. In *Computer Graphics International (CGI'11 electronic proceedings)*, June 2011.

[16] T.C. Lee, R.L. Kashyap, and C.N. Chu. Building skeleton models via 3-D medial surface/axis thinning algorithms. *Graphical Models and Image Processing*, 56(6):462–478, November 1994.

[17] P. Lindstrom. Out-of-core simplification of large polygonal models. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 259–262, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[18] Jean-Luc Mari. Surface sketching with a voxel-based skeleton. In *15th IAPR International Conference on Discrete Geometry for Computer Imagery (DGCI'09)*, volume 5810 of *Lecture Notes in Computer Science*, pages 325–336. Springer, 2009.

[19] M. Meyer, M. Desbrun, P. Schroeder, and A.H. Barr. Discrete differential geometry operators for triangulated 2-manifolds. *VisMath*, 2002.

[20] J. Rossignac and P. Borrel. Multi-resolution 3D approximation for rendering complexe scences. *Geometric Modeling In Computer Graphics*, pages 455–465, 1993.

[21] Christian Rössl, Leif Kobbelt, and Hans-Peter Seidel. Extraction of feature lines on triangulated surfaces using morphological operators. In *AAAI Spring Symposium on Smart Graphics*, volume 00-04, pages 71–75, March 2000.

[22] E. Shaffer and M. Garland. Efficient adaptative simplification of massive meshes. *In Proceedings of IEEE Visualization 2001*, pages 127–134, October 2001.

[23] Kaleem Siddiqi and Stephen Pizer. *Medial Representations. Mathematics, Algorithms and Applications*. Computational Imaging and Vision, Vol. 37. Springer, 2008.

[24] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. *Fifth International Conference on Computer Vision*, pages 902–907, 1995.

[25] W. T. Vertterling, S. A Teukolsky, W. H. Press, and B. P. Flannery. *Numerical Recipe in C/C++, The Art of Scientific Computing*. 2003.

[26] G. Xu. Convergence analysis of a discretization scheme for gaussian curvature over triangular surfaces. *Comput. Aided Geom. Des.*, 23(2):193–207, 2006.

[27] Kai Yu, Jiangqin Wu, and Yueting Zhuang. Skeleton-based recognition of chinese calligraphic character image. In *Advances in Multimedia Information Processing (PCM'08)*, volume 5353 of *Lecture Notes in Computer Science*, pages 228–237. Springer, 2008.

[28] T. Y. Zhang and C. Y. Suen. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3):236–239, March 1984.

Aix-Marseille University, LSIS, UMR CNRS 7296, Marseille, France • alexandra.bac@univ-amu.fr • jean-luc.mari@univ-amu.fr • viseur@cerege.fr • marc.daniel@univ-amu.fr

Tangential Approximation of Surfaces

Carl OLSSON and Yuri BOYKOV

Abstract

In the Computer Vision community it is a common belief that higher order smoothness, such as curvature, should be modeled using higher order interactions. For example, 2nd order derivatives for deformable (active) contours are represented by triple cliques. Similarly, the 2nd order regularization methods in stereo predominantly use MRF models with scalar (1D) disparity labels and triple clique interactions. In this paper we give an overview of an energy minimization framework for *tangential approximation of surfaces* developed in [21, 22]. The framework uses higher dimensional labels to encode second order smoothness with pairwise interactions. Hence, many generic optimization algorithms (e.g. message passing, graph cut, etc.) can be used to optimize the proposed regularization functionals. The accuracy of our approach for representing curvature is demonstrated by theoretical and empirical results on real data sets from multi-view reconstruction and stereo.

1. INTRODUCTION

Surface estimation from point measurements is an important problem in Computer Vision. This paper gives an overview of an energy minimization framework for *tangential approximation of surfaces* developed in [21, 22]. The approach assumes that we are given noisy position estimates/probabilities of points sampled from the surface. To each point measurement our method assigns a tangent plane that represents the exact surface position and orientation. The utilized measurements are application specific and could, for example, be 3D-positions (from multiple view geometry or laser scans), 2D projections, or photoconsistency (from image data). Our energy includes a regularization term that encourages tangents to agree on some underlying piece-wise smooth surface by estimating 2nd order smoothness (such as curvature or 2nd derivative).

Our framework is based on a graphical model. In general, graphical models are widely used in vision for problems like dense stereo [30, 5], surface estimation [29], image segmentation [5], inpainting [25], etc. Perhaps the most common regularizer is *length* (or *area* in 3D). This smoothness criteria corresponds to graphical models with simple pair-wise potentials, which admit very efficient global optimization algorithms like [5, 15].

Recently, the vision community has begun to actively explore models using curvature-based regularization or similar second-order smoothness priors [25, 9, 26, 30]. Typically, evaluation of the second-order smoothness properties of a curve/surface requires an interaction between three or more points. This leads to hard-to-optimize graphical models with higher-order cliques instead of simple pairwise potentials.

In [21, 22] we take an alternative approach. Instead of using graphical models with higher-order interactions, we extend our label space so that the labels (tangents) encode both position and orientation of a curve/surface. We therefore get an easier to handle pairwise interaction at the expense of a larger search space. This makes it possible to evaluate curvature using only pairwise interactions. Thus, standard combinatorial optimization methods, e.g. TRW-S [15], can be readily applied. The idea of simplifying optimization problems by lifting the label space has also been applied in the context of variational formulations for optical flow and segmentation [20, 23, 6]. To

---

demonstrate out general approach we consider two applications; Point Cloud Regularization and Stereo Reconstruction.

1.1. Regularization of Point Clouds.

Point clouds obtained from structure-from-motion type procedures [11] typically contain significant amounts of noise. One reason for this is that there is no regularization term that encourages point positions to agree on smooth surfaces in the final solution. To regularize the point cloud we estimate a "true" position and a tangent plane, for each point, such that neighboring points agree on an underlying piecewise smooth surface. The tangent interaction that we use can be shown to estimate normal curvature of the surface. Our approach does not assume any particular topology, nor does it have to be closed or "orientable".

Piece-wise constant regularization has previously been applied to similar geometric problems, e.g. [13, 8]. Their labels are global geometric primitives (lines, circles, homographies). For a general scene such an approach may require enumerating a very large number of primitives to account for all possible surfaces in a scene. In contrast, this paper adopts an approach where global non-parametric surfaces are formed by smoothly combining locally estimated primitives (e.g. tangent lines/planes). As shown by [28] in the context of non-rigid structure from motion, similar *local-to-global* approach may work even without regularization. In general, however, piece-wise smooth MRF priors [3] are necessary to build global surfaces from local primitives estimated from ambiguous data with noise and outliers.

In the graphics community surface estimation from point clouds is widely studied [12]. Popular approaches are least squares surfaces (MLS) [1], locally optimal projections (LOP) [19], and anisotropic point cloud diffusion [16]. However, none of these methods regularize curvature. Perhaps, the closest to our approach is [27] that uses active surface elements, a.k.a. *surfels*. In principle, curvature could be estimated from pairwise potentials between the surfels. However, this is difficult in practice since surfels correspond to a label space with six degrees of freedom (point position and a local coordinate frame). In contrast, our labels have only three d.o.f. (tangent plane).

1.2. Stereo Reconstruction.

The goal of stereo reconstruction is to compute a depth estimate for every pixel in an image. Doing so requires that every pixel in the image is matched to a corresponding pixel in another image. Due to ambiguous texture this matching is rarely unique and as a result the stereo problem is most often ill posed. Resolving these ambiguities requires adding knowledge of what types of surfaces that we can expect to see in natural scenes, in the form of regularization.

First order methods [5, 14, 10] often implicitly assume fronto-parallel planes. For example, standard piecewise smooth (e.g. truncated linear or quadratic) pairwise regularization potentials assign higher cost to surfaces with larger tilt with respect to the camera [5]. To model surfaces more accurately Birchfeld and Tomasi [2] introduced 3D-labels corresponding to arbitrary 3D planes. However, this approach is limited to piecewise planar scenes. To address more general scenes recent papers use 2nd derivative regularization [22, 18, 30]. There are two ways of modeling such higher order smoothness potentials. Woodford et al. [30] retain the scalar disparity labels while using triple-cliques to penalize 2nd derivatives of the reconstructed surface. This encourages near planar smooth disparity maps. The optimization problem is however made more difficult due to the introduction of non-submodular triple interactions. In contrast, [22, 18] use 3D-labels corresponding to tangent planes to encode 2nd order smoothness as pairwise interactions. It is shown in [22] that in contrast to the triple-cliques used by Woodford et al. [30] the 3D-label formulation is often submodular (or near submodular) making fusion moves easier to solve optimally using standard methods like Roof duality [24].

1.3. Optimization Background.

In this section we briefly review the concept of fusion moves [17]. Given two (possibly very different) solution proposals the fusion move tries to compute the best possible combination, with respect to the objective function, by selecting the best parts from each proposal. This allows modifying a large number of pixels simultaneously thereby escaping poor local minima.

Consider the optimization of an arbitrary second order pseudo-boolean function (PBF) of n variables, usually expressed as,

$$(1.1) \qquad \min_{\boldsymbol{x} \in \boldsymbol{L}^n} E(\boldsymbol{x}) = \min_{\boldsymbol{x} \in \boldsymbol{L}^n} \sum_{p \in \mathcal{V}} U_p(x_p) + \sum_{(p,q) \in \mathcal{E}} V_{pq}(x_p, x_q)$$

where $\boldsymbol{L} = \{0, 1\}$. We think of the objective function as being defined on a graph $G = (\mathcal{V}, \mathcal{E})$. To each node $p \in \mathcal{V}$ we want to assign a binary value, at a cost specified by the unary term U_p. In addition, for each edge $(p, q) \in \mathcal{E}$ there is a term V_{pq} that specifies a costs for combinations of assignments to p and q. If V_{pq} is submodular, that is

$$(1.2) \qquad V_{pq}(0, 0) + V_{pq}(1, 1) \leq V_{pq}(0, 1) + V_{pq}(1, 0),$$

this can be efficiently solved [5]. Even with V_{pq} non-submodular roof-duality (RD) [4, 24] can be used. RD will give a partial solution where labeled variables are guaranteed to be correct for an optimal solution and some variables are left *unlabeled*.

Lempitsky et al. [17] proposed a way to minimize (1.1) when $\boldsymbol{L} = \mathbb{R}$. Given two assignments $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ we fuse them into a new one with lower energy by solving

$$(1.3) \qquad \min_{\boldsymbol{z} \in \{0,1\}^n} E(\boldsymbol{z} \cdot \boldsymbol{x}_0 + (\boldsymbol{1} - \boldsymbol{z}) \cdot \boldsymbol{x}_1),$$

where $\cdot$ is element-wise multiplication. If we solve (1.3) using RD and then set $\boldsymbol{z} = \boldsymbol{0}$ for all unlabeled variables the autarky results in [24] gives

$$(1.4) \qquad E(\boldsymbol{z} \cdot \boldsymbol{x}_0 + (\boldsymbol{1} - \boldsymbol{z}) \cdot \boldsymbol{x}_1) \leq \min\big(E(\boldsymbol{x}_0), E(\boldsymbol{x}_1)\big).$$

Therefore we can iteratively minimize (1.1) by proposing new solutions and fusing them with the old solution.

The possibility to decrease the energy for each fusion move is an attractive feature, however there is no guarantee on how many variables will be labeled in each fusion move. For submodular fusion moves we are guaranteed to label all variables. Minimizing a submodular function is also faster in practice [24].

2. An Energy Approach to Tangential Estimation

Next we present the approach for tangent estimation. Recall the general formulation of our problem: Given noisy point position estimates/probabilities we want to estimate the true point positions by enforcing smoothness of the unknown underlying surface. Note that estimating 2nd order smoothness of a surface from point samples requires at least 3 points; first derivatives are estimated using pairs of points then second derivatives using pairs of pairs. To avoid triple interactions like these our labels encode both point position and derivatives using a tangent plane. In this way we are able to encode our smoothness using pairwise interactions at the expense of having to search a larger label space.

We can think of the method as an energy minimization problem on a graph $G = (\mathcal{V}, \mathcal{E})$. To each node $p \in \mathcal{V}$ we will assign a tangent $\boldsymbol{t}_p = (\mathbf{n}_p, a_p)$, where $\mathbf{n}_p \in \mathbb{R}^3$, $||\mathbf{n}_p|| = 1$ and $a_p \in \mathbb{R}$. Note that we assume that each tangent also provides (a unique) estimate of the sought point position $\boldsymbol{p}$. For example, in Section 3 $\boldsymbol{p}$ will be the point on the tangent that has the smallest distance to the point measurement. This limits the dimension of our labels to 3. If this is not the case then in addition to the tangent itself we would have to estimate a position on the plane requiring 2 additional degrees of freedom.

The energy has the form

$$(2.1) \qquad E(\boldsymbol{t}) = \sum_{(p,q) \in \mathcal{E}} V_{pq}(\boldsymbol{t}_p, \boldsymbol{t}_q) + \sum_{p \in \mathcal{V}} U_p(\boldsymbol{t}_p).$$

The unary term U_p only depends on the tangent assigned to p. This term is sometimes refereed to as data term since it is constructed from point measurements/probabilities. In Section 3 we will use the distance between the estimated and the measured point and in Section 4 it will be based on photo-consistency.

The pairwise term V_{pq} provides the regularization. This term will encourage assignments where neighboring nodes have similar tangents. Here the neighborhood is defined by the edge set $\mathcal{E}$. Specifically the term V_{pq} works by considering the effect of switching the assigned tangent at q
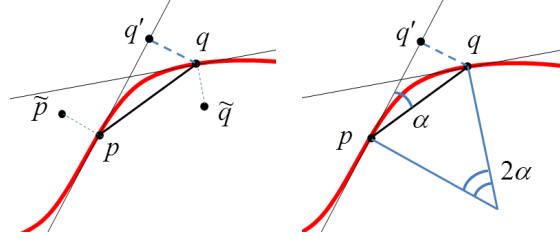
Figure 3.1. Left: pairwise interaction approximating curvature corresponds to quotient (3.12). Right: the quotient $\frac{|q-q'|}{|p-q|^2}$ yields half the curvature at p under the assumption that p and q belong to a constant curvature segment.

to its neighboring tangent $\boldsymbol{t}_p$. Intuitively, if the surface is smooth, then the tangents at nearby nodes p and q should be similar. Therefore the point estimations should not change much when switching tangents. An alternative way of thinking of this type of interaction is as the deviation from the 1st order approximation at p (tangent $\boldsymbol{t}_p$) measured at q, which is a measure of the 2nd order properties of the surface. Due to the restriction to pairwise potentials we need to measure surface smoothness indirectly by estimating smoothness of curves on the surface. More complex regularization terms like Gaussian curvature require higher order potentials.

Note that we do not explicitly construct any surface. It is only locally implicitly represented using the tangents. Therefore our approach does not assume any particular surface topology, nor does it have to be closed or "orientable". Furthermore, in contrast to methods that estimate smoothness using explicit polygonal approximations, such as triangulations, we do not need to know in which order the points are connected. For each node p we just compute smoothness estimates using all points in a small neighborhood.

3. POINT CLOUD REGULARIZATION

We assume that points $\tilde{\boldsymbol{p}}$ and $\tilde{\boldsymbol{q}}$ are noisy measurements of the points $\boldsymbol{p}$ and $\boldsymbol{q}$ on a curve on the underlying surface (see Figure 3.1). To $\tilde{\boldsymbol{p}}$ and $\tilde{\boldsymbol{q}}$ we assign tangents $(\mathbf{n}_p, a_p)$ and $(\mathbf{n}_q, a_q)$. The estimated "true" points $\boldsymbol{p}$ and $\boldsymbol{q}$ will be the orthogonal projections of $\tilde{\boldsymbol{p}}$ and $\tilde{\boldsymbol{q}}$ onto the assigned tangents, that is

$$\boldsymbol{p} = \tilde{\boldsymbol{p}} - (\mathbf{n}_p^\top \tilde{\boldsymbol{p}} + a_p)\mathbf{n}_p \tag{3.1}$$

$$\boldsymbol{q} = \tilde{\boldsymbol{p}} - (\mathbf{n}_q^\top \tilde{\boldsymbol{q}} + a_q)\mathbf{n}_q. \tag{3.2}$$

Given the estimations $\boldsymbol{p}$ and $\boldsymbol{q}$ and their tangents we would like to estimate how smooth the underlying curve is. There are several ways of doing this. Bruckstein et al. [7] use angles between consecutive line segments to measure curvature. While this gives an easy type of interaction it requires an explicit representation of the estimated polygonal curve. In addition it does not seem possible to generalize this measure to the 3-dimensional case without resorting to triple cliques. In contrast we are seeking an implicit local estimation using tangents with pairwise interactions. To measure the difference between the two tangent planes we therefore compute the projections (see Fig.3.1 left)

$$\boldsymbol{p}' = \boldsymbol{p} - (\mathbf{n}_q^\top \boldsymbol{p} + a_q)\mathbf{n}_q \tag{3.3}$$

$$\boldsymbol{q}' = \boldsymbol{q} - (\mathbf{n}_p^\top \boldsymbol{q} + a_p)\mathbf{n}_p, \tag{3.4}$$

that is, the estimated points projected onto the neighboring tangent plane. If the underlying surface is smooth then

$$|\boldsymbol{p}' - \boldsymbol{p}| + |\boldsymbol{q}' - \boldsymbol{q}| \tag{3.5}$$

should be small since the tangents are close to each other. The term (3.5) is closely related to curvature. To see this we assume that $(\mathbf{n}_p, a_p)$, $(\mathbf{n}_q, a_q)$ are the true tangent planes at the points

$\boldsymbol{p}$ and $\boldsymbol{q}$, and that

(3.6)
$$\boldsymbol{p} = \alpha(t_1)$$

(3.7)
$$\boldsymbol{q} = \alpha(t_2),$$

where α is a smooth curve. Furthermore, we will assume that this curve is parametrized by arc-length. (This is however no restriction since any curve with derivative $\dot{\alpha} \neq 0$ can be re-parametrized.) Since $\mathbf{n}_p^\top \boldsymbol{p} + a_p = 0$ we have

(3.8)
$$|\boldsymbol{q} - \boldsymbol{q}'| = |\mathbf{n}_p^\top \boldsymbol{q} + a_p| = |\mathbf{n}_p^\top (\boldsymbol{q} - \boldsymbol{p})|$$

Using the Taylor expansion of α at t_1

(3.9)
$$\alpha(t) = \alpha(t_1) + \dot{\alpha}(t_1)(t - t_1) + \frac{1}{2}\ddot{\alpha}(t_1)(t - t_1)^2 + O((t - t_1)^3),$$

together with (3.6)-(3.7), the term (3.8) can be written

(3.10)
$$|\mathbf{n}_p^\top \left(\dot{\alpha}(t_1)(t_2 - t_1) + \frac{1}{2}\ddot{\alpha}(t_1)(t_2 - t_1)^2 \right) + O((t - t_1)^3)|.$$

Since $\dot{\alpha}(t_1)$ is the tangent at $\boldsymbol{p}$ it is perpendicular to the normal $\mathbf{n}_p$ so the first term vanishes. Furthermore, since α is parametrized by arc length, $\ddot{\alpha}$ is perpendicular to the tangent $\dot{\alpha}$. Therefore $\mathbf{n}_p^\top \ddot{\alpha}(t_1)$ is only the length of $\ddot{\alpha}(t_1)$ (plus or minus, depending on the direction of $\mathbf{n}_p$), which is the curvature. Hence if we divide $|\boldsymbol{q} - \boldsymbol{q}'|$ by $(t_2 - t_1)^2$ we get roughly half the curvature. However, as we estimate the underlying curve (or any parametrization of it) t_1 and t_2 are unknown. Again using the Taylor expansion we note that

(3.11)
$$|\boldsymbol{p} - \boldsymbol{q}|^2 = |\dot{\alpha}(t_1)|(t_2 - t_1)^2 + O((t_2 - t_1)^3)$$

Since α is a unit speed curve $|\dot{\alpha}| = 1$, and therefore

(3.12)
$$\frac{|\boldsymbol{q} - \boldsymbol{q}'|}{|\boldsymbol{p} - \boldsymbol{q}|^2} = \frac{\frac{1}{2}|\mathbf{n}_p^\top \ddot{\alpha}(t_1) + O(t_2 - t_1)|}{|1 + O(t_2 - t_1)|}.$$

This expression will tend to half of the curvature when t_2 tends to t_1, and therefore approximates curvature well if $\boldsymbol{p}$ and $\boldsymbol{q}$ are close enough. Note that if we think of the normals as coming form an underlying surface then the interaction can be interpreted as measuring normal curvature of the surface in the direction $\boldsymbol{q}' - \boldsymbol{p}$.

Figures 3.2 shows the computation of the quotient (3.12) for the curve $\sqrt{3}\cos(t)$. To approximate the integral of the absolute curvature we use

(3.13)
$$\frac{1}{2}\int |\kappa| d\sigma \approx \sum_i \frac{|\boldsymbol{p}_{i+1} - \boldsymbol{p}'_{i+1}|}{|\boldsymbol{p}_{i+1} - \boldsymbol{p}_i|},$$

where $\boldsymbol{p}_i$ are the sampled 2D points. Since the function has amplitude $\sqrt{3}$ the derivative is $\pm\sqrt{3}$ in its endpoints. This gives 60 degree angles to the x-axis at the endpoints, which is very close to what the approximations in Figures 3.2 give.

An interesting special case where the approximation turns out to be exact is when the points are lying on a constant curvature segment, that is, a segment that is part of a circle or line. For simplicity let us assume that $\boldsymbol{p}$ and $\boldsymbol{q}$ is on a circle with center at the origin, see Figure 3.1 right. If we parametrize the curve by arc length the angle 2α is t/r. By the cosine theorem we get

(3.14)
$$|\boldsymbol{p} - \boldsymbol{q}|^2 = 2r^2(1 - \cos(\frac{t}{r}))$$

Since the angle α is $t/2r$ we see from the second triangle that

(3.15)
$$|q - q'| = |p - q'|\sin(\frac{t}{2r})$$

Assuming $0 \leq \frac{t}{2r} \leq \pi$, this gives us the quotient

(3.16)
$$\frac{|\boldsymbol{q} - \boldsymbol{q}'|}{|\boldsymbol{p} - \boldsymbol{q}|^2} = \frac{\sin(\frac{t}{2r})}{\sqrt{2r^2(1 - \cos(\frac{t}{r}))}} = \frac{\sin(\frac{t}{2r})}{2r|\sin(\frac{t}{2r})|} = \frac{1}{2r},$$

which is half of the curvature of a circle with radius r.
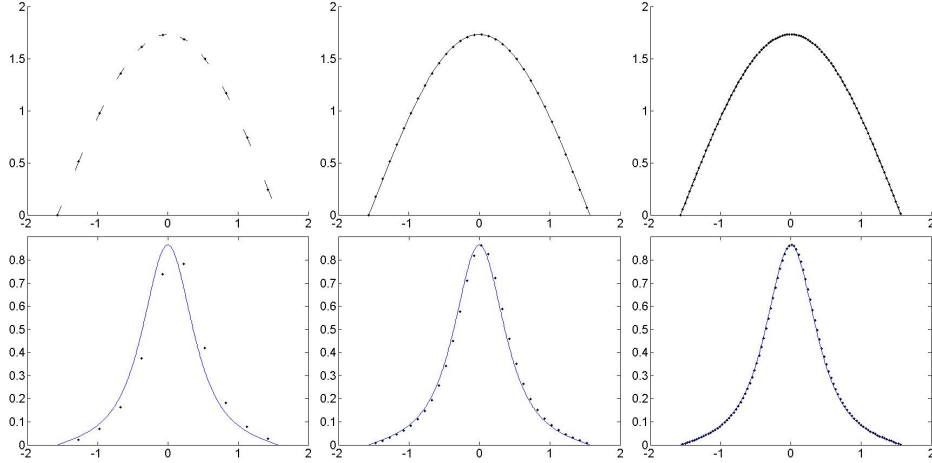
Figure 3.2. Computation of the curvature of $\sqrt{3}\cos(t)$, $-\pi/2 \leq t \leq \pi/2$. Top: The sample points and their tangent planes for various sampling rates. Bottom: The computed curvature at the sample points, and the true curvature of the function. The values of the integral approximations are 59.6265, 59.9600 and 59.9961 degrees respectively (converted from radians).

3.1. **Experiments.** In this section we present some experimental results. For the energy we use (2.1) with

$$(3.17) \qquad V_{pq}(\boldsymbol{t}_p, \boldsymbol{t}_q) = \min\left(\frac{|\boldsymbol{q} - \boldsymbol{q}'|}{|\boldsymbol{p} - \boldsymbol{q}|}, \tau\right),$$

where τ is a threshold that ensures that we do no over penalize transitions between surfaces. Note that we consider the edges of $\mathcal{E}$ to be undirected. Therefore if V_{pq} is present in the energy then so is V_{qp} and therefore the energy is symmetric. Since all nodes do not necessarily have the same number of neighbors we weight the pairwise interaction V_{pq} with one over the number of neighbors of p.

The unary term is simply the squared distance $U_p(\boldsymbol{t}_p) = |\tilde{\boldsymbol{p}} - \boldsymbol{p}|^2$. For further details see [21].

We apply it to the two real datasets depicted in figures 3.3 and 3.4. The first one is a set of 3D points at the surface of a castle (18270 points), and the second one is one of the laser scans of the well known Stanford bunny (40256 points) The first dataset is more noisy since it was created using a 3D reconstruction scheme, whereas the bunny was scanned in a laboratory setting. Figure 3.3 and 3.4 shows the obtained results. The settings used where $\mu = 5e5$, $\tau = 0.56$ for the bunny, and and $\mu = 2500$, $\tau = 0.56$ for the castle. In both cases the solution took roughly 3 hours to compute.

To generate proposals for the fusion moves we use an approach similar to that of [8, 13]. First candidate tangents are computed using random sampling. These are then refined using local optimization of the energy and fused with the current solution.

For the computations we used the code available from [15] and the Matlab wrapper from [30]. Since our energy is not of any of the standard forms supported by [15], we used the general lookup table form (which is highly inefficient according [15]) to setup our potentials. Specialized software should therefore be able to speed up these computations considerably.

4. STEREO RECONSTRUCTION

In dense stereo the objective is to compute a depth $z(p)$ (distance to the surface/object from the image) for each pixel p in an images. The result is an estimate of the surface geometry for all points on the object that are visible in the image.

The cost of assigning a particular depth to a pixel is based on comparison of pixel appearances. From the pixel coordinate and the given depth it is possible to compute the position of the corresponding point in a second image. Comparing the appearances (e.g. RGB-values) of these two pixels gives a cost of that depth. Individual depth estimates are usually unreliable due
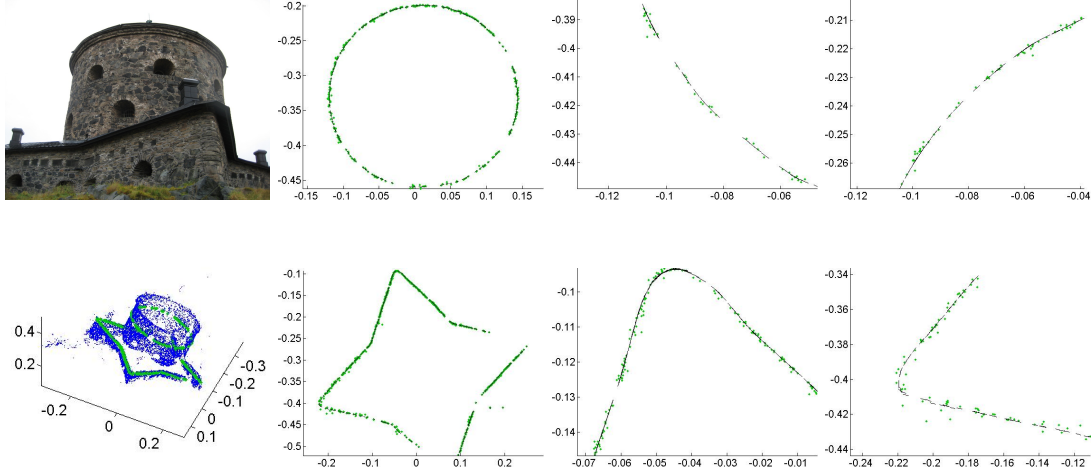
Figure 3.3. Visualization of the resulting tangent planes along two planar cuts. *1st column*: Image of the scene, Input 3D points (blue, 18270 in total) and points (green) close to the two cutting planes. *2nd column*: Points (green) projected onto the plane and line segments (black) obtained when intersecting the corresponding tangent planes with the cutting plane. *3rd and 4th columns*: Two closeups.
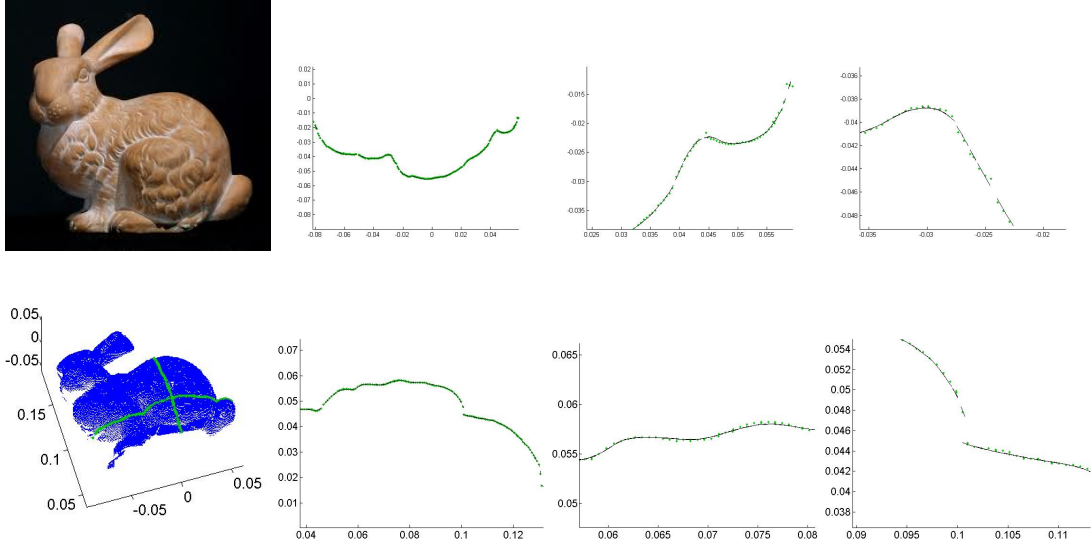
Figure 3.4. Visualization of the resulting tangent planes along two planar cuts. *1st column*: Image of the scene, Input 3D points (blue, 40256 in total) and points (green) close to the two cutting planes. *2nd column*: Points (green) projected onto the plane and line segments (black) obtained when intersecting the corresponding tangent planes with the cutting plane. *3rd and 4th columns*: Two closeups.

to noise and ambiguous texture and we therefore need to apply regularization. However directly penalizing 2nd derivative of the depth function is not a good idea. In general the projection of a plane will not yield a linear depth function unless the camera is affine (which can be seen from (4.4) below). Hence, such an energy would assign a 3D-plane a nonzero penalty. Therefore we will instead measure the deviation from the tangent plane along the viewing ray.

Let $\boldsymbol{p}_h$ and $\boldsymbol{q}_h$ denote the homogeneous coordinates (with third coordinate 1) and $\boldsymbol{p}$ and $\boldsymbol{q}$ regular Cartesian coordinates of the two pixels p and q. We will assume a pinhole camera model where the center camera has been calibrated and normalized to be of form $[\boldsymbol{I}\ \boldsymbol{0}]$. Given a function $z : \mathcal{I} \mapsto \mathbb{R}_+$ that gives a depth for each pixel the 3D points $\boldsymbol{P}$ and $\boldsymbol{Q}$ corresponding to p and q can
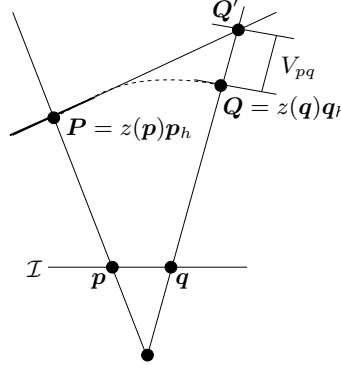
Figure 4.1. Left: Geometric interpretation of the stereo smoothness interaction.

be computed (in regular Cartesian coordinates) using the simple formulas

$$P = z(p)p_h, \tag{4.1}$$

$$Q = z(q)q_h. \tag{4.2}$$

By $t_p = (\mathbf{n}_p, a_p)$, where $\mathbf{n}_p \in \mathbb{R}^3$, $\|\mathbf{n}_p\| = 1$ and $a_p \in \mathbb{R}$ we denote the tangent plane at p given by the equation

$$\mathbf{n}_p^T x + a_p = 0. \tag{4.3}$$

Consider the intersection point Q' between the viewing ray at q and the tangent plane at p (see Figure 4.1). We let $\mathcal{T}_p z : \mathcal{I} \mapsto \mathbb{R}_+$ be the depth function of the tangent plane at point p, that is, $Q' = \mathcal{T}_p z(q) q_h$. We can calculate the tangent function using

$$\mathcal{T}_p z(q) = -\frac{a_p}{\mathbf{n}_p^T q_h}. \tag{4.4}$$

(Here we are assuming that the viewing ray is not completely contained in the tangent plane.) Note that even though this function represents a plane in 3D it is usually not linear in q.

To encourage smooth assignments we use the cost

$$V_{pq}(t_p, t_q) = \|Q - Q'\| = |\mathcal{T}_p z(q) - z(q)| \, \|q_h\|. \tag{4.5}$$

The geometric interpretation of this expression can be seen in Figure 4.1. Given the estimated tangent plane at p and the depth at q the interaction computes the distance between the estimated 3D point and the tangent plane along the viewing ray. The smoothness term will penalize deviations from planes and thereby encourage solutions with small second derivatives. Similarly to the previous section it can be shown that this interaction will penalize

$$\left| \frac{(z')^2 - zz''}{2z} \right|. \tag{4.6}$$

The term $(z')^2/z$ compensates for the fact that viewing rays are not parallel in the pinhole camera model. (In case of parallel rays the interaction just approximates $|z''|$.) The reason for not using the same interaction as in Section 3 for stereo is illustrated in Figure 4.2. The interaction from Section 3 can be interpreted as fitting a constant curvature segment between p and q with tangent t_p. This only works well if p and q are close (recall that p and q were 3D points in Section 3), since this is when the Taylor expansion is valid. As can be seen in Figure 4.2 for large depth differences the curvature will be significantly underestimated. In contrast the interaction presented in this section can be interpreted as an approximation with $\frac{(z')^2 - zz''}{2z}$ which is a more reasonable approximation. In addition it is valid as long as the pixel coordinates p and q are close and therefore it is not sensitive to large depth differences between pixels.

4.1. **Experiments.** Next we evaluate the proposed framework on a couple of multiple view stereo data sets. For the energy we use

$$V_{pq}(t_p, t_q) = \min(Q - Q', \tau). \tag{4.7}$$
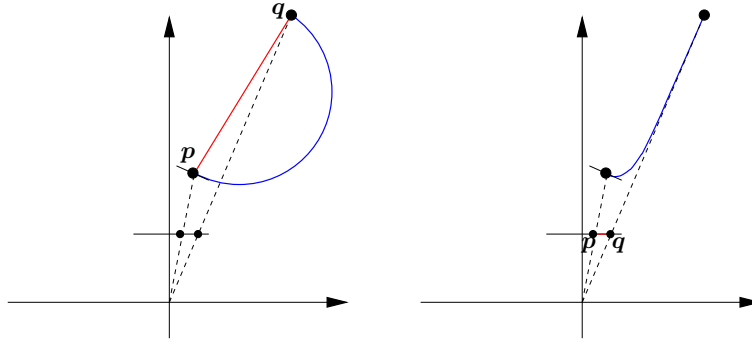
Figure 4.2. Differences between the two interactions in Section 3 and Section 4.
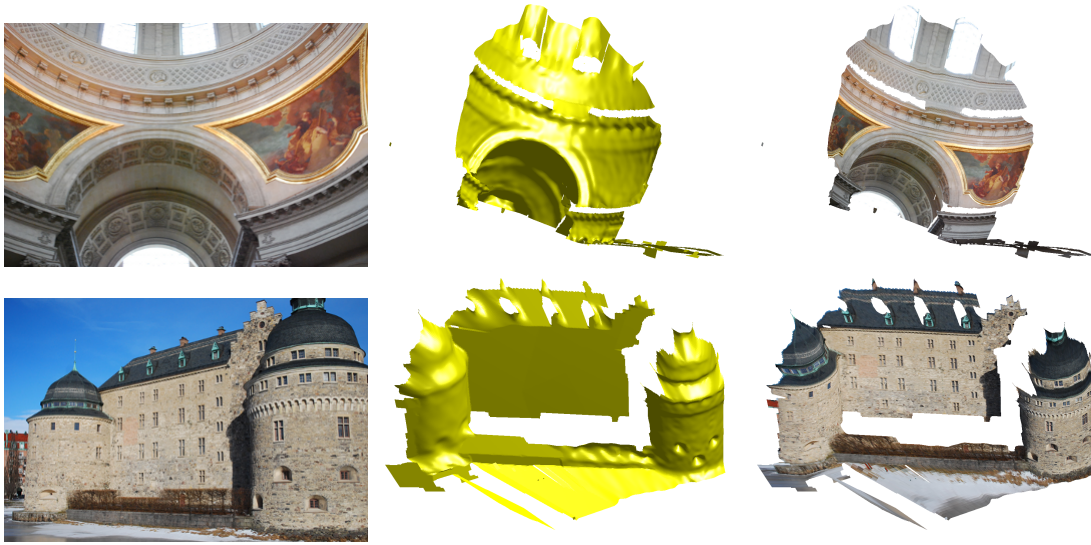
Figure 4.3. Resulting surface estimations. Left: Image. Middle: Estimated surface. Right: Estimated surface with texture.

For the data term U_p we use normalized cross correlation (NCC) computed at different possible depths. Quadratic interpolation is used to compute values in between the sampled depths. See [22] for further details on implementations.

To generate proposals we use random sampling of planes and local optimization. In the sampling step we select a pixel and a small neighborhood around it. Using the best local maximum of the normalized cross correlation for each viewing ray in the neighborhood we create a 3D cloud and fit a plane using RANSAC. Fusion moves with these planar proposals are particularly effective and can in fact be shown to be submodular [22].

Figure 4.3 shows the result of applying this approach to two datasets.

References

[1] Marc Alexa, Johannes Behr, Daniel Cohen-or, Shachar Fleishman, David Levin, and Claudio T. Silva. Computing and rendering point set surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 9:3–15, 2003.

[2] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *International Conference on Computer Vision*, 1999.

[3] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, Cambridge, USA, 1987.

[4] E. Boros and P.L. Hammer. Pseudo-boolean optimization. *Discrete applied mathematics*, 123(1):155–225, 2002.

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2001.

[6] Kristian Bredies, Thomas Pock, and Benedikt Wirth. Convex relaxation of a class of vertex penalizing functionals. *Journal of Mathematical Imaging and Vision*, pages 1–25, 2012.

[7] A.M. Bruckstein, A.N. Netravali, and T.J. Richardson. Epi-convergence of discrete elastica. *Applicable Analysis*, 79(1-2):137–171, 2001.

[8] Andrew Delong, Anton Osokin, Hossam Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, 96(1):1–27, January 2012.

[9] Noha El-Zehiry and Leo Grady. Fast global optimization of curvature. In *Proc. of CVPR 2010*. IEEE Computer Society, IEEE, June 2010.

[10] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. *Int. J. Comput. Vision*, 70(1):41–54, October 2006.

[11] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[12] H. Huang, D. Li, Hao Zhang, Uri Ascher, and Daniel Cohen-Or. Consolidation of unorganized point clouds for surface reconstruction. In *ACM Trans. on Graphics*, 2009.

[13] Hossam Isack and Yuri Boykov. Energy-Based Geometric Multi-model Fitting. *International Journal of Computer Vision*, 97(2):123–147, April 2012.

[14] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *European Conf. on Computer Vision*, volume III, pages 82–96, Copenhagen, Denmark, 2002.

[15] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28:1568–1583, October 2006.

[16] Carsten Lange and Konrad Polthier. Anisotropic smoothing of point sets. *Computer Aided Geometric Design*, 22:2005, 2005.

[17] V. S. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion moves for markov random field optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1392–1405, 2010.

[18] G. Li and S.W. Zucker. Differential geometric inference in surface stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(1):72–86, 2010.

[19] Yaron Lipman, Daniel Cohen-Or, David Levin, and Hillel Tal-Ezer. Parameterization-free projection for geometry reconstruction. In *ACM Trans. om Graphics.*, 2007.

[20] Tal Nir, Alfred M. Bruckstein, and Ron Kimmel. Over-parameterized variational optical flow. *Int. J. Comput. Vision*, 76(2):205–216, February 2008.

[21] C. Olsson and Y. Boykov. Curvature-based regularization for surface approximation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[22] C. Olsson, J. Ulén, and Y. Boykov. In defense of 3d-label stereo. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[23] Guy Rosman, Shachar Shem-tov, David Bitton, Tal Nir, Gilad Adiv, Ron Kimmel, Arie Feuer, and Alfred M. Bruckstein. Over-parameterized optical flow using a stereoscopic constraint. In *SSVM*, 2011.

[24] C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *IEEE conf. on Computer Vision and Pattern Recognition*, 2007.

[25] T. Schoenemann, F. Kahl, and D. Cremers. Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *Int. Conf. on Computer Vision*, Kyoto, Japan, 2009.

[26] Petter Strandmark and Fredrik Kahl. Curvature regularization for curves and surfaces in a global optimization framework. In *EMMCVPR*, pages 205–218, 2011.

[27] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling surfaces of arbitrary topology with dynamic particles. In *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR '93., 1993 IEEE Computer Society Conference on*, 1993.

[28] Jonathan Taylor, Allan D. Jepson, and Kiriakos N. Kutulakos. Non-rigid structure from locally-rigid motion. In *IEEE Int. Conf. of Computer Vision and Pattern Recognition*, San Francisco, 2010.

[29] George Vogiatzis, Carlos Hernández Esteban, Philip H. S. Torr, and Roberto Cipolla. Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(12):2241–2246, 2007.

[30] O.J. Woodford, P.H.S. Torr, I.D. Reid, and A.W. Fitzgibbon. Global stereo reconstruction under second order smoothness priors. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

Centre for Mathematical Sciences, Lund University, SWEDEN • calle@maths.lth.se

Computer Science Department, University of Western Ontario, CANADA • yuri@csd.uwo.ca

An overview of curvature bounds and spectral theory of planar tessellations

Matthias KELLER

Abstract

We give a survey about the spectral consequences of upper bounds on the curvature on planar tessellating graphs. We first discuss spectral bounds and then put a particular focus on uniformly decreasing curvature. This case is characterized by purely discrete spectrum and we further present eigenvalue asymptotics and exponential decay of eigenfunctions. We then discuss absence of compactly supported eigenfunctions and dependence of the spectrum of the Laplacian on the underlying ℓ^p space.

1. INTRODUCTION

In this article we survey results relating curvature bounds and spectral theory. We focus on infinite planar tessellations which can be considered as discrete analogues of non compact surfaces. The tiles of the tessellations are all treated as regular polygons.

We study a curvature function which arises as an angular defect. This notion of curvature is justified by the Gauß Bonnet formula. This idea goes back at least to Descartes, see [16], and appeared since then independently at various places, see e.g. [37, 20, 25, 40]. A substantial amount of research was conducted to study the geometric property of the tessellation in dependence of the curvature, see e.g. [3, 4, 5, 9, 10, 11, 22, 23, 24, 26, 29, 34, 37, 39, 40, 42]. The operator under investigation is the graph Laplacian on the tessellation with constant edge weights. First, we show spectral bounds resulting from curvature bounds. Here, the quantitative bounds result from estimates in [29]. Secondly, we take a closer look at the case of uniformly unbounded negative curvature. This case characterizes discreteness of spectrum, [26], for which we present eigenvalue asymptotics, [6, 7], and decay properties of eigenfunctions, [28]. Thirdly, unique continuation properties of eigenfunctions are discussed, [7, 30], and, finally we summarize results on the p-dependence of the spectrum of the Laplacian as an operator on ℓ^p, $p \in [1, \infty]$.

2. SET UP AND DEFINITIONS

In this section we introduce planar tessellations, notions of curvature and the graph Laplacian.

2.1. **Planar tessellations.** Let (V, E) be a simple planar graph embedded into an orientable topological surface $\mathcal{S}$ that is homeomorphic to $\mathbb{R}^2$. We assume the embedded graph is *locally finite*, i.e., for every compact $K \subseteq \mathcal{S}$, one has

$$\#\{e \in E \mid e \cap K \neq \emptyset\} < \infty.$$

In particular, this excludes the situation that a vertex has infinitely many neighbors. In the following we do not distinguish between the graph and its embedding. Nevertheless, we stress that we only use the combinatorial properties of the graph which do not depend on the embedding.

The set of *faces* F has the connected components of

$$\mathcal{S} \setminus \bigcup E$$

as elements. For $f \in F$, we denote by $\overline{f}$ the closure of f in $\mathcal{S}$. We denote $G = (V, E, F)$. Following [3, 4], we call $G = (V, E, F)$ a *tessellation* if the following three assumptions are satisfied:

(T1) Every edge is contained in two faces.
(T2) Two faces are either disjoint or intersect in a vertex or an edge.
(T3) Every face is homeomorphic to the unit disc.

There are related definitions such as semi-planar graphs see [23, 24] and locally tessellating graphs [27]. Indeed, most of the results presented here hold for general planar graphs on surfaces of finite genus. However, the definition of curvature becomes more involved and some of the estimates turn out to me more technical. Thus, we stick to the 'tame' special case of tessellations.

2.2. **Curvature.** In order to define a curvature function, we first introduce the notation for the vertex degree and the face degree. We denote the *vertex degree* of a vertex $v \in V$ by

$$|v| = \#\text{edges emanating from } v$$

and the *face degree* of a face $f \in F$ by

$$|f| = \#\text{boundary edges of } f = \#\text{boundary vertices of } f.$$

The *vertex curvature* $\kappa : V \to \mathbb{R}$ is defined as

$$\kappa(v) = 1 - \frac{|v|}{2} + \sum_{f \in F, v \in \overline{f}} \frac{1}{|f|}.$$

The idea traces back at least to Descartes [16] and was later introduced in the above form by Stone in [37] referring to ideas of Alexandrov. Since then this notion of curvature was widely used, see e.g. [3, 4, 11, 22, 24, 26, 27, 29, 34, 40, 42]. The notion of curvature is motivated as an angular defect: Assume a face f is a regular polygon. Then, the inner angles of f are all equal to

$$\beta(f) = 2\pi \frac{|f| - 2}{2|f|}.$$

This formula is easily derived as walking around f once results in an angle of 2π, while going around the $|f|$ corners of f one takes a turn by an angle of $\pi - \beta(f)$ each time. In this light the vertex curvature might be rewritten as

$$2\pi\kappa(v) = 2\pi - \sum_{f \in F, v \in \overline{f}} \beta(f), \quad v \in V.$$

Nevertheless, it should be stressed that the mathematical nature of κ is purely combinatorial while assuming a particular nice embedding it allows for a geometric interpretation. The notion has its further justification in the Gauß Bonnet formula which is mathematical folklore and may for instance be found in [3] or [27].

A finer notion of curvature arises when one asks which contribution to the total curvature at a vertex v comes from a corner at a face f with $v \in \overline{f}$. Precisely, the set of *corners* of a tessellation G is given by

$$C(G) = \{(v, f) \in V \times F \mid v \in \overline{f}\}.$$

Define the *corner curvature* $\kappa_C : C(G) \to \mathbb{R}$ by

$$\kappa_C(v, f) = \frac{1}{|v|} - \frac{1}{2} + \frac{1}{|f|}.$$

One immediately infers

$$\kappa(v) = \sum_{f \in F, v \in \overline{f}} \kappa_C(v, f).$$

This notion of curvature was first introduced in [3] and further studied in [4, 27].

2.3. **The Laplacian.** Next, we come to the graph Laplacian. Consider the quadratic form $Q : C(V) \to [0, \infty]$

$$Q(f) = \frac{1}{2} \sum_{v \sim w} |f(v) - f(w)|^2,$$

where $C(V)$ is the space of complex valued functions on V. Choosing the counting measure on V yields a Hilbert space $\ell^2(V)$ of complex valued functions whose absolute value square is summable. The scalar product on $\ell^2(V)$ is given by

$$\langle f, g \rangle = \sum_{v \in V} \overline{f}(v) g(v), \quad f, g \in \ell^2(V),$$

and the norm by $\|f\| = \langle f, f \rangle^{\frac{1}{2}}$. Restricting Q to the subspace

$$\{f \in \ell^2(V) \mid Q(f) < \infty\}$$

yields a positive quadratic form which can seen to be closed. By general theory, there is a positive selfadjoint operator Δ associated to Q which acts as

$$\Delta f(v) = \sum_{w \sim v} (f(v) - f(w))$$

and has the domain

$$D(\Delta) = \{f \in \ell^2(V) \mid \Delta f \in \ell^2(V)\}.$$

The complex valued functions of compact support $C_c(V)$ are dense in $D(\Delta)$, see [41]. It is not hard to see that the operator Δ is bounded if and only if

$$\sup_{v \in V} |v| < \infty.$$

Remark 2.1. There is another common choice of a measure on V, namely the vertex degree function. This way one determines the volume of a set by counting the number of edges (twice the number of edges with both end vertices in the set and ones the number of edges having only one vertex in the set) rather than by the number of vertices in the case of the counting measure. Restricting the quadratic form Q to the corresponding ℓ^2 space yields a different positive selfadjoint operator Δ_n which is always bounded and is often referred to as the normalized Laplacian.

3. The bottom of the spectrum

In this section we turn to the bottom of the spectrum of Δ. In the case where the bottom of the spectrum is strictly positive one speaks of existence of a spectral gap. We discuss that non-negative curvature implies absence of a spectral gap while in the case of negative curvature we show existence of a spectral gap for which we then present quantitative estimates.

Since the operator Δ is a positive selfadjoint operator on the Hilbert space $\ell^2(V)$, its spectrum $\sigma(\Delta)$ is included in the positive half axis $[0, \infty)$. We consider the bottom of the spectrum of Δ

$$\lambda_0(\Delta) = \inf \sigma(\Delta)$$

which, by the Rayleigh-Ritz variational characterization, is equal to

$$\lambda_0(\Delta) = \inf_{f \in D(\Delta), \|f\|=1} \langle f, \Delta f \rangle = \inf_{f \in C_c(V), \|f\|=1} \langle f, \Delta f \rangle,$$

where the second equality follows from the density of $C_c(V)$ in $D(\Delta)$.

3.1. **Non-negative curvature.** The following theorem is a rather immediate consequence of considerations of Jost/Hua/Liu in [24, Theorem 4.1], that every non-negatively curved planar graph has at most quadratic volume growth, and [13, 17, 35], that the bottom of spectrum of graphs with subexponential volume is zero which holds for general graphs. An important technicality is that a lower curvature bound implies boundedness of Δ and, thus, [13, 17, 35] are applicable by [26, Theorem 1].

Theorem 3.1. *Assume $\kappa \geq 0$, then $\lambda_0(\Delta) = 0$.*

3.2. **Average negative curvature.** In [40], Woess showed that the isoperimetric constant of a planar tessellation is strictly positive whenever the curvature is negative on average on large sets. He defined

$$\overline{\kappa} = \limsup_{n \to \infty} \inf_{W \subseteq V, n \le |W| < \infty} \frac{1}{|W|} \sum_{v \in W} \kappa(v).$$

From [40, Theorem 1] and [26, Theorem 1] we infer the next result.

Theorem 3.2. *Assume $\overline{\kappa} < 0$, then $\lambda_0(\Delta) > 0$.*

Remark 3.3. Dodziuk proved in [12] that planar graphs with $|v| \ge 7$, $v \in V$, satisfy $\lambda_0(\Delta) > 0$. In particular, this assumption implies $\kappa < 0$. Independently to the theorem above, but somewhat later, Higuchi [22] showed that $\kappa < 0$ implies $\lambda_0(\Delta) > 0$.

3.3. **Negative curvature.** The theorem above only yields positivity of the bottom of the spectrum. In the case of negative curvature, we get the following quantitative result which is a direct consequence of [29, Theorem 1], [18, Proposition 1] and estimate [26, Theorem 1].

Theorem 3.4. *Assume $\kappa < 0$ and let*

$$K = -\sup_{v \in V} \frac{\kappa(v)}{|v|} \quad and \quad d = \min_{v \in V} |v|.$$

Then there is a constant $C \ge 1$ specified below such that

$$\lambda_0(\Delta) \ge d(1 - \sqrt{1 - 4C^2 K^2}) \ge 2dC^2 K^2,$$

where for $p = \sup_{v \in V} |v|$ and $q = \sup_{f \in F} |f|$ the constant C is given by

$$C := \begin{cases} 1 & : \ if \ q = \infty, \\ 1 + \frac{2}{q-2} & : \ if \ q < \infty \ and \ p = \infty, \\ (1 + \frac{2}{q-2})(1 + \frac{2}{(p-2)(q-2)-2}) & : \ if \ p, q < \infty. \end{cases}$$

Remark 3.5. (a) The second inequality in the theorem follows simply by the Taylor expansion of the square root.

(b) The theorem above can be considered as a discrete analogue to a theorem of McKean [32] who proves for a n-dimensional complete Riemannian manifold M with upper sectional curvature bound $-k$ that the bottom of the spectrum of the Laplace-Beltrami $\Delta_M \ge 0$ satisfies

$$\lambda_0(\Delta_M) \ge (n-1)^2 k/4.$$

(c) A curious fact noted by Higuchi [22], see also [43], is that if $\kappa < 0$, then already $\kappa \le -1/1806$ which is the case that a triangle, a heptagon and a 43-gon meet in a vertex. Indeed, this implies that $\kappa < 0$ yields for the constant in our theorem $K > 0$ and, therefore, $\lambda_0(\Delta) > 0$.

4. DISCRETE SPECTRUM, EIGENVALUE ASYMPTOTICS AND DECAY OF EIGENFUNCTIONS

In this section we study the case of uniformly decreasing curvature. That is if the quantity

$$\kappa_\infty = \inf_{K \subseteq V \text{ finite}} \sup_{v \in V \setminus K} \kappa(v)$$

equals $-\infty$. In this case, we discuss below that the spectrum of Δ consists only of discrete eigenvalues which accumulate at ∞. We denote the eigenvalues in increasing order counted with multiplicity by $\lambda_n(\Delta)$, $n \ge 0$. Moreover, we discuss the asymptotics of the eigenvalues and the exponential decay of eigenfunctions.

4.1. **Discrete spectrum.** The next theorem which is characterizing pure discrete spectrum is found in [26, Theorem 3].

Theorem 4.1. *The spectrum of Δ is purely discrete if and only if $\kappa_\infty = -\infty$.*

Remark 4.2. (a) The theorem above can be considered as a discrete analogue of a theorem of Donnelly/Li [15] which states that a negatively curved, complete Riemannian manifold M with sectional curvature bound decaying uniformly to $-\infty$ the Laplace-Beltrami operator Δ_M has pure discrete spectrum.

(b) In [18] Fujiwara proved a related statement for the normalized Laplacian on trees, namely that spectrum is discrete except for the point 1, where the discrete eigenvalues accumulate.

(c) Wojciechowski [41] showed also discreteness of the spectrum of Δ on general graphs in terms of a different quantity which is sometimes referred to as a mean curvature.

4.2. **Eigenvalue asymptotics.** An important observation in the proof of the theorem above is the following estimate

$$-\frac{|v|}{2} \leq \kappa(v) \leq 1 - \frac{|v|}{6}, \quad v \in V.$$

This inequality implies that $|\cdot|$ and $-\kappa$ go simultaneously to ∞.

In particular, if $\kappa_\infty = -\infty$, then there is a bijective map $\mathbb{N}_0 \to V$, $n \mapsto v_n$, such that

$$|v_n| \leq |v_{n+1}|, \qquad n \geq 0.$$

The following eigenvalue asymptotics are found in the recent work [7] improving the results for planar graphs which were obtained in [19] for trees and for general graphs in [6] .

Theorem 4.3. *If $\kappa_\infty = -\infty$, then*

$$|v_n| - 2\sqrt{|v_n|} \lesssim \lambda_n \lesssim |v_n| + 2\sqrt{|v_n|},$$

that is

$$\lim_{n\to\infty} \frac{\lambda_n}{|v_n|} = 1$$

and

$$-1 \leq \liminf_{n\to\infty} \frac{\lambda_n - |v_n|}{2\sqrt{|v_n|}} \leq \limsup_{n\to\infty} \frac{\lambda_n - |v_n|}{2\sqrt{|v_n|}} \leq 1.$$

Remark 4.4. The only related results we are aware of are found in [33] for the adjacency matrix on sparse finite graphs.

4.3. **Decay of eigenfunctions.** After having treated eigenvalues, we next come to the decay of eigenfunctions. It turns out that eigenfunctions decay exponentially in an ℓ^2 sense, see [28].

Theorem 4.5. *If $\kappa_\infty - \infty$ and $\varphi_n \in D(\Delta)$, $n \geq 0$, are eigenfunctions, i.e.,*

$$\Delta\varphi_n = \lambda_n\varphi_n$$

then, for any $\beta < e^{-1}$ and $o \in V$,

$$|\kappa|^{\frac{1}{2}} e^{\beta d(o,\cdot)}\varphi_n \in \ell^2(V),$$

where $d(\cdot,\cdot)$ is the natural graph distance.

Remark 4.6. The proof is based on ideas based on the work of Agmon for Schrödinger operators in $\mathbb{R}^d$. The somewhat curious constant e^{-1} comes in via an optimization that is caused by the non-locality of the graph Laplacian in contrast to the strongly local Laplace operator on $\mathbb{R}^n$.

5. Unique continuation of eigenfunctions

In Riemannian manifolds very strong unique continuation properties of eigenfunctions hold. Often very subtle quantitative statements can be obtained which all have the basic corollary that there are no eigenfunctions of compact support. However, for graphs such eigenfunctions can be produced rather easily, see e.g. [1, 8, 14, 31] for examples. In this section we discuss that having non-positive corner curvature excludes such eigenfunctions.

5.1. **Absence of compactly supported eigenfunctions.** Klassert/Lenz/Peyerimhoff/Stollmann [30] proved the following unique continuation result for tessellations with non-positive corner curvature, $\kappa_C \leq 0$, which was later generalized to planar graphs in [27] with a different proof.

Theorem 5.1. *If $\kappa_C \leq 0$, then there are no eigenfunctions of compact support.*

We stress that the assumption $\kappa_C \leq 0$ can not be relaxed to $\kappa \leq 0$ (or $\kappa < 0$). This can be seen by the example that contains a $2n$-gon f and has triangles attached at every edge f. Now, given a function which takes the values ± 1 alternating around the vertices f and zero otherwise, can be seen to be a compactly supported eigenfunction to the eigenvalue 6.

5.2. **Finitely many compactly supported eigenfunctions.** One can now ask whether the exponentially decaying eigenfunctions of Theorem 4.5 can be compactly supported. As it can be seen from the theorem above if $\kappa_C \leq 0$ then there are no such eigenfunctions at all. However, $\kappa_\infty = -\infty$ only implies $\kappa_C \leq 0$ outside of a finite set. The following theorem, found in [7], tells us that in this case there can be at most finitely many linearly independent compactly supported eigenfunctions.

Theorem 5.2. *If $\kappa_\infty = -\infty$, then there is a finite set such that every eigenfunction of compact support is supported in this set.*

It can be seen from the proof in [7] that $\kappa_\infty = -\infty$ is not necessary but it is sufficient to have sufficiently negative curvature outside of a finite set. On the other hand, it is also shown in [7] that $\kappa_C \leq 0$ outside of a finite set is not enough.

6. The ℓ^p spectrum

Finally we come to the spectrum of the Laplacian as an operator on $\ell^p(V)$, $p \in [1, \infty]$. For these Banach spaces the Laplacian Δ_p acts as Δ and has the domain

$$D(\Delta_p) = \{\varphi \in \ell^p(V) \mid \Delta\varphi \in \ell^p(V)\}.$$

The operators Δ_p are the generators of the extension of the semigroup $e^{-t\Delta_2}$, $t > 0$, to $\ell^p(V)$, $p \in [1, \infty)$, and Δ_∞ is the adjoint of Δ_1. An important question which was initially brought up by Simon [36] and answered by Hempel/Voigt [21] for Schrödinger operators is whether the spectrum depends on the underlying Banach space. Sturm, [38], addressed this question in the manifold settings in terms of uniform subexponential volume growth. A special case he considers is the case of curvature bounds. An analogous result for graphs is proven in [2]. As a consequence of this theorem and some geometric and functional analytic ingredients one can derive the following theorem which is also found in [2].

Theorem 6.1.   (a) *If $\kappa \geq 0$, then $\sigma(\Delta_2) = \sigma(\Delta_p)$ for all $p \in [1, \infty]$.*
  (b) *If $-K \leq \kappa < 0$, then $\lambda_0(\Delta_2) \neq \lambda_0(\Delta_1)$.*
  (c) *If $\kappa_\infty = -\infty$, then $\sigma(\Delta_2) = \sigma(\Delta_p)$ for all $p \in (1, \infty)$.*

References

[1] Michael Aizenman and Simone Warzel. The canopy graph and level statistics for random operators on trees. *Math. Phys. Anal. Geom.*, 9(4):291–333 (2007), 2006.

[2] Frank Bauer, Bobo Hua, and Matthias Keller. On the l^p spectrum of Laplacians on graphs. *Adv. Math.*, 248:717–735, 2013.

[3] O. Baues and N. Peyerimhoff. Curvature and geometry of tessellating plane graphs. *Discrete Comput. Geom.*, 25(1):141–159, 2001.

[4] Oliver Baues and Norbert Peyerimhoff. Geodesics in non-positively curved plane tessellations. *Adv. Geom.*, 6(2):243–263, 2006.

[5] Ethan D. Bloch. A characterization of the angle defect and the Euler characteristic in dimension 2. *Discrete Comput. Geom.*, 43(1):100–120, 2010.

[6] Michel Bonnefont, Sylvain Golénia, and Matthias Keller. Eigenvalue asymptotics for Schrödinger operators on sparse graphs. *preprint*, 2013.

[7] Michel Bonnefont, Sylvain Golénia, and Matthias Keller. Eigenvalue asymptotics and unique continuation of eigenfunctions on planar graphs. *preprint*, 2014.

[8] Jonathan Breuer and Matthias Keller. Spectral analysis of certain spherically homogeneous graphs. *Oper. Matrices*, 7(4):825–847, 2013.

[9] Beifang Chen. The Gauss-Bonnet formula of polytopal manifolds and the characterization of embedded graphs with nonnegative curvature. *Proc. Amer. Math. Soc.*, 137(5):1601–1611, 2009.

[10] Beifang Chen and Guantao Chen. Gauss-Bonnet formula, finiteness condition, and characterizations of graphs embedded in surfaces. *Graphs Combin.*, 24(3):159–183, 2008.

[11] Matt DeVos and Bojan Mohar. An analogue of the Descartes-Euler formula for infinite graphs and Higuchi's conjecture. *Trans. Amer. Math. Soc.*, 359(7):3287–3300 (electronic), 2007.

[12] Jozef Dodziuk. Difference equations, isoperimetric inequality and transience of certain random walks. *Trans. Amer. Math. Soc.*, 284(2):787–794, 1984.

[13] Jozef Dodziuk and Leon Karp. Spectral and function theory for combinatorial Laplacians. In *Geometry of random motion (Ithaca, N.Y., 1987)*, volume 73 of *Contemp. Math.*, pages 25–40. Amer. Math. Soc., Providence, RI, 1988.

[14] Józef Dodziuk, Peter Linnell, Varghese Mathai, Thomas Schick, and Stuart Yates. Approximating L^2-invariants and the Atiyah conjecture. *Comm. Pure Appl. Math.*, 56(7):839–873, 2003. Dedicated to the memory of Jürgen K. Moser.

[15] Harold Donnelly and Peter Li. Pure point spectrum and negative curvature for noncompact manifolds. *Duke Math. J.*, 46(3):497–503, 1979.

[16] Pasquale Joseph Federico. *Descartes on polyhedra*, volume 4 of *Sources in the History of Mathematics and Physical Sciences*. Springer-Verlag, New York-Berlin, 1982. A study of the ıt De solidorum elementis.

[17] Koji Fujiwara. Growth and the spectrum of the Laplacian of an infinite graph. *Tohoku Math. J. (2)*, 48(2):293–302, 1996.

[18] Koji Fujiwara. The Laplacian on rapidly branching trees. *Duke Math. J.*, 83(1):191–202, 1996.

[19] Sylvain Golénia. Hardy inequality and asymptotic eigenvalue distribution for discrete Laplacians. *J. Funct. Anal.*, 266(5):2662–2688, 2014.

[20] M. Gromov. Hyperbolic groups. In *Essays in group theory*, volume 8 of *Math. Sci. Res. Inst. Publ.*, pages 75–263. Springer, New York, 1987.

[21] Rainer Hempel and Jürgen Voigt. The spectrum of a Schrödinger operator in $L_p(\mathbf{R}^\nu)$ is p-independent. *Comm. Math. Phys.*, 104(2):243–250, 1986.

[22] Yusuke Higuchi. Combinatorial curvature for planar graphs. *J. Graph Theory*, 38(4):220–229, 2001.

[23] Bobo Hua and Jürgen Jost. Geometric analysis aspects of infinite semiplanar graphs with nonnegative curvature II. *to appear in Trans. Amer. Math. Soc.*

[24] Bobo Hua, Jürgen Jost, and Shiping Liu. Geometric analysis aspects of infinite semiplanar graphs with nonnegative curvature. *to appear in J. Reine Angew. Math.*

[25] M. Ishida. Pseudo-curvature of a graph. *lecture at 'Workshop on topological graph theory', Yokohama National University*, 1990.

[26] Matthias Keller. The essential spectrum of the Laplacian on rapidly branching tessellations. *Math. Ann.*, 346(1):51–66, 2010.

[27] Matthias Keller. Curvature, geometry and spectral properties of planar graphs. *Discrete Comput. Geom.*, 46(3):500–525, 2011.

[28] Matthias Keller and Daniel Lenz. Agmon type estimates and purely discrete spectrum for graphs. *preprint.*

[29] Matthias Keller and Norbert Peyerimhoff. Cheeger constants, growth and spectrum of locally tessellating planar graphs. *Math. Z.*, 268(3-4):871–886, 2011.

[30] Steffen Klassert, Daniel Lenz, Norbert Peyerimhoff, and Peter Stollmann. Elliptic operators on planar graphs: unique continuation for eigenfunctions and nonpositive curvature. *Proc. Amer. Math. Soc.*, 134(5):1549–1559, 2006.

[31] Steffen Klassert, Daniel Lenz, and Peter Stollmann. Discontinuities of the integrated density of states for random operators on Delone sets. *Comm. Math. Phys.*, 241(2-3):235–243, 2003.

[32] H. P. McKean. An upper bound to the spectrum of Δ on a manifold of negative curvature. *J. Differential Geometry*, 4:359–366, 1970.

[33] Bojan Mohar. Many large eigenvalues in sparse graphs. *European J. Combin.*, 34(7):1125–1129, 2013.

[34] Byung-Geun Oh. Duality Properties of Strong Isoperimetric Inequalities on a Planar Graph and Combinatorial Curvatures. *Discrete Comput. Geom.*, 51(4):859–884, 2014.

[35] Yoshiki Ohno and Hajime Urakawa. On the first eigenvalue of the combinatorial Laplacian for a graph. *Interdiscip. Inform. Sci.*, 1(1):33–46, 1994.

[36] Barry Simon. Brownian motion, L^p properties of Schrödinger operators and the localization of binding. *J. Funct. Anal.*, 35(2):215–229, 1980.

[37] David A. Stone. A combinatorial analogue of a theorem of Myers. *Illinois J. Math.*, 20(1):12–21, 1976.

[38] Karl-Theodor Sturm. On the L^p-spectrum of uniformly elliptic operators on Riemannian manifolds. *J. Funct. Anal.*, 118(2):442–453, 1993.

[39] Liang Sun and Xingxing Yu. Positively curved cubic plane graphs are finite. *J. Graph Theory*, 47(4):241–274, 2004.

[40] Wolfgang Woess. A note on tilings and strong isoperimetric inequality. *Math. Proc. Cambridge Philos. Soc.*, 124(3):385–393, 1998.

[41] Radoslaw Krzysztof Wojciechowski. *Stochastic completeness of graphs*. ProQuest LLC, Ann Arbor, MI, 2008. Thesis (Ph.D.)–City University of New York.

[42] Lili Zhang. A result on combinatorial curvature for embedded graphs on a surface. *Discrete Math.*, 308(24):6588–6595, 2008.

[43] Andrzej Żuk. On the norms of the random walks on planar graphs. *Ann. Inst. Fourier (Grenoble)*, 47(5):1463–1490, 1997.

Mathematisches Institut , Friedrich Schiller Universität Jena , 07743 Jena, Germany • m.keller@uni-jena.de

Generalized Ricci curvature and the geometry of graphs

Frank Bauer, Bobo Hua, Jürgen Jost, and Shiping Liu

1. Curvature

Originally, curvature was a concept of differential geometry developed with the purpose of describing the geometry of surfaces in space in a manner analogous to those of curves. Something is curved if it is not straight like a line or flat like a plane. Gauss [10] then realized that such a notion of curvature of surfaces in space confused two different aspects. One is concerned with how the surface bends in space, that is, how its normal direction changes when moving along the surface. The other, in contrast, is concerned with the inner geometry of that surface, that is, for instance, how slowly or fast geodesic curves emanating from the same point in different directions move away from each other. Riemann [18] then developed an intrinsic geometry on manifolds of arbitrary dimension built upon such an intrinsic curvature, see [12] for the current state of this important field of mathematics. The Riemann sectional curvature measures such a divergence of geodesics whose initial directions all lie in the same tangent plane. Averaging the curvatures over all such planes containing a given direction v then yields the Ricci curvature in the direction v. Finally, averaging over all directions v starting at the same point p yields the scalar curvature at p. In this contribution, we shall mainly be concerned with the properties of Ricci curvature. Ricci curvature characterizes the growth of the volume of distance balls as a function of their radius. More precisely, Ricci curvature controls the cost of transporting the mass of one distance ball to another one. When the Ricci curvature is large, the volumes of balls become smaller, but the relative volumes of the intersection of two balls become larger. Therefore, such a transport becomes less costly. Ricci curvature also yields lower bounds for the first nonzero eigenvalue of the Laplace operator on a compact Riemannian manifold.

While these curvature concepts were originally developed for Riemannian manifolds, that is, differentiable manifolds equipped with a smooth metric tensor, the characteristic properties of curvature just described are meaningful for more general metric spaces. Therefore, notions of generalized sectional or Ricci curvature have been developed that are meaningful for certain classes of metric spaces that are more general than Riemannian manifolds.

In particular, such concepts then also apply to graphs. For instance, one can consider an undirected and unweighted graph G as a metric space with each edge isometric to the unit interval, that is, of length one. For each vertex x, one also has a natural probability measure m_x on G that assigns the weight $\frac{1}{d_x}$ to every neighbor of x, where d_x is the degree of x, that is, the number of its neighbors, the vertices connected to x by an edge. All other vertices, including x itself, get the weight 0 under m_x. Again, we can ask for the cost of transporting m_x to m_y when x and y are neighbors. Ollivier [16] then defined Ricci curvature bounds for graphs in terms of such transportation costs.

In this contribution, we shall explain how the generalized Ricci curvature as defined by Ollivier relates to other characteristic properties of graphs, like the clustering coefficient [19] that is important for the analysis of social and other networks. We also show how this generalized Ricci

curvature controls the smallest as well as the largest eigenvalue of the normalized graph Laplacian. In fact, we obtain nontrivial eigenvalue estimates for all graphs that are not bipartite. Our constructions utilize the concept of the neighborhood graph [4], a geometric representation of the concept of a random walk on a graph. Thereby, we see a natural link between Ricci curvature, eigenvalues, and stochastic analysis.

While these principles hold in more generality, here we only explore them for graphs.

2. Generalized Ricci curvature

Ollivier's [16, 17] definition of Ricci curvature depends on the L^1-Wasserstein distance.

Definition 2.1. Let (X, d) be a metric space equipped with its Borel sigma algebra, and let m_1, m_2 be probability measures on X. The L^1-Wasserstein or transportation distance between the probability measures m_1 and m_1 is

$$(2.1) \qquad W_1(m_1, m_1) = \inf_{\xi^{x,y} \in \prod(m_1, m_2)} \sum_{(x', y') \in V \times V} d(x', y') \xi^{x,y}(x', y'),$$

where $\prod(m_1, m_2)$ is the set of probability measures $\xi^{x,y}$ that satisfy

$$(2.2) \qquad \sum_{y' \in V} \xi^{x,y}(x', y') = m_1(x'), \ \sum_{x' \in V} \xi^{x,y}(x', y') = m_2(y').$$

The conditions (2.2) mean that we start with the measure m_1 and end up with m_2. When we consider the distance $d(x', y')$ as the transportation cost from x' to y', then $W_1(m_1, m_2)$ is the minimal cost to transport the mass of m_1 to that of m_2. $\xi^{x,y}$ is considered as a transfer plan between m_1 and m_2, or a coupling of the two random walks governed by m_1 and m_2, respectively. Those $\xi^{x,y}$ which attain the infimum in (2.1) are called optimal couplings.

The transportation distance $W_1(m_1, m_2)$ can also expressed by the Kantorovich duality formula,

$$(2.3) \qquad W_1(m_1, m_2) = \sup_{f : \text{Lip}(f) \leq 1} \left[\sum_{x' \in V} f(x') m_1(x') - \sum_{y' \in V} f(y') m_2(y') \right],$$

where $\text{Lip}(f) := \sup_{x \neq y} \frac{|f(x) - f(y)|}{d(x,y)}$ is the Lipschitz seminorm of f.

Definition 2.2. Let (X, d) be a complete and separable metric space equipped with its Borel sigma algebra and a family of probability measueres $m_x, x \in M$ which depend measurably on x and which have finite first moments, i.e., $\int_M d(x, y) dm_x(y) < \infty$. For any two distinct points $x, y \in M$, the (Ollivier-) Ricci curvature of (X, d, m) then is defined as

$$(2.4) \qquad \kappa(x, y) := 1 - \frac{W_1(m_x, m_y)}{d(x, y)}.$$

The probability measures m_x could also be interpreted as the probability densities associated to a random walk, that is, $m_x(y)$ is the probability that a random walker at x jumps to y in one time step.

Here, we shall restrict our attention to graphs considered as metric spaces with the measures m_x explained in Section 1.

3. Ricci curvature and the geometry of graphs

3.1. **Basic notions from graph theory.** We introduce some basic definitions and constructions from graph theory, including the (normalized) graph Laplacian, see [13] and the references given there.

We first consider a locally finite unweighted graph $G = (V, E)$. V is the vertex and E the edge set. We say that $x, y \in V$ are neighbors, and write $x \sim y$, when they are connected by an edge. The degree d_x of a vertex x is defined as the number of its neighbors.

We also assume that G is connected, that is, for every pair of distinct vertices $x, y \in V$, there exists a path between them, that is, a sequence $x = x_0, x_1, \ldots, x_m = y$ of distinct vertices such that $x_{\nu-1} \sim x_\nu$ for $\nu = 1, \ldots, m$. (Not connected graphs can simply be decomposed into their connected components.) A cycle in G is a closed path $x_0, x_1, \ldots, x_m = x_0$ for which all the vertices $x_1, \ldots, x_m$ are distinct. For $m = 3, 4, 5, \ldots$, we speak of a triangle, quadrangle, pentagon,... A graph without

cycles is called a tree. A graph is called bipartite if its vertex set can be decomposed into two disjoint components V_1, V_2 such that whenever $x \sim y$, then x and y are in different components. Any tree is bipartite. More generally, a graph is bipartite iff it has no cycles of odd length. In particular, it has no triangles.

To get a metric, for neighbors x, y, we put $d(x, y) = 1$. For arbitrary vertices x, y, $d(x, y)$ is the length of the shortest path connecting x and y, i.e. the minimal number of edges that needs to be traversed to get from x to y.

We next introduce the (normalized) graph Laplacian operating on L^2-functions on the vertex set V, see e.g. [8, 13]. Here, we use the scalar product

$$(3.1) \qquad (v, u) := \sum_{x \in V} d_x v(x) u(x)$$

to define $L^2(G)$. We then put

$$\Delta : L^2(G) \to L^2(G)$$

$$(3.2) \qquad \Delta v(x) := \frac{1}{d_x} \left(\sum_{y, y \sim x} v(y) - d_x v(x) \right) = \frac{1}{d_x} \sum_{y, y \sim x} v(y) - v(x).$$

This is a discrete analogue of the Laplace-Beltrami operator of a Riemannian manifold. We can also consider, for neighbors $x \sim y$, the discrete differential

$$(3.3) \qquad Du(x, y) := u(y) - u(x),$$

a discrete analogue of the differential of a function. D can be considered as a map from functions on the vertices of D to functions on the edges of D. In order to make the latter space also an L^2-space, we introduce the product

$$(3.4) \qquad (Du, Dv) := \sum_{e=(x,y)} (u(y) - u(x))(v(y) - v(x)).$$

Note that we are summing here over edges, and not over vertices. If we did the latter, we would need to put in a factor $1/2$ because each edge would then be counted twice. We then have

$$(3.5) \qquad (\Delta u, v) = -(Du, Dv)$$

for all $u, v \in L^2(G)$.

We now list some basic properties of Δ.

(1) Δ is selfadjoint w.r.t. $(., .)$:

$$(3.6) \qquad (u, \Delta v) = (\Delta u, v)$$

for all $u, v \in L^2(G)$. This follows from (3.5).

(2) Δ is nonpositive:

$$(3.7) \qquad (\Delta u, u) \leq 0$$

for all u. This follows from the Cauchy-Schwarz inequality.

(3) $\Delta u = 0$ iff u is constant. In fact, when $\Delta u = 0$, there can neither be a vertex x with $u(x) \geq u(y)$ for all $y \sim x$ with strict inequality for at least one such y, since $\Delta u(x) = 0$ means that the value $u(x)$ is the average of the values at the neighbors of x. Since G is assumed to be connected, u then has to be a constant (if G were not connected, a solution of $\Delta u = 0$ would have to be constant on every connected component of G.) Of course, this is a discrete version of the standard maximum principle argument.

We are interested in the eigenvalues of the Laplacian, that is, in those λ with

$$(3.8) \qquad \Delta u + \lambda u = 0$$

for some nontrivial function $u \in L^2(G)$, called an eigenfunction for λ. From the properties of Δ just listed, we can infer some immediate consequences for the eigenvalues.

- All eigenvalues are real, because Δ is selfadjoint.
- All eigenvalues are nonnegative, because Δ is a nonpositive operator.

- The smallest eigenvalue is $\lambda_0 = 0$, with a constant eigenfunction. Since we assume that Γ is connected, this eigenvalue is simple. In other words,

(3.9) $$\lambda_k > 0$$

for $k > 0$ where we order the eigenvalues as

$$\lambda_0 = 0 < \lambda_1 \leq ... \leq \lambda_K$$

and put $K := N - 1$.
- The largest eigenvalue λ_{N-1} is 2 iff G is bipartite and is < 2 else. (See [4] for details and a systematic analysis of the highest eigenvalue.)

The eigenfunctions v_i, v_j for different eigenvalues λ_i, λ_j are orthogonal to each other,

(3.10) $$(v_i, v_j) = 0.$$

In particular, since the constants are the eigenfunctions for the eigenvalue $\lambda_0 = 0$, for all $i > 0$, we then have

(3.11) $$\sum_x m_x v_i(x) = 0.$$

3.2. **Ricci curvature and clustering.** In this section, we essentially describe the results of [14]. As explained, in order to define Ricci curvature, we need the probability measures from Section 1

(3.12) $$m_x(y) = \begin{cases} \frac{1}{d_x} & \text{if } y \sim x; \\ 0 & \text{otherwise.} \end{cases}$$

We can interpret this in terms of a random walker that sits at x at time $t \in \mathbb{N}$ and then selects a neighbor of x with equal probability $\frac{1}{d_x}$ as the target of his walk at time $t + 1$. .

Theorem 3.1. *On a locally finite graph $G = (V, E)$, we have for any pair of neighboring vertices x, y,*

$$\kappa(x, y) \geq - \left(1 - \frac{1}{d_x} - \frac{1}{d_y} - \frac{\sharp(x, y)}{d_x \wedge d_y} \right)_+ - \left(1 - \frac{1}{d_x} - \frac{1}{d_y} - \frac{\sharp(x, y)}{d_x \vee d_y} \right)_+ + \frac{\sharp(x, y)}{d_x \vee d_y},$$

where we have put

$$d_x \wedge d_y := \min\{d_x, d_y\}, \quad d_x \vee d_y := \max\{d_x, d_y\}.$$

Remark: For the case where $\sharp(x, y) = 0$, this result was obtained in [15]. For our purposes, however, the key point is to understand how the presence of triangles in a graph improves the lower Ricci bound.

The proof of Theorem 3.1 depends on a careful transport plan, according to the definition of Ricci curvature. We do not present the details, but the following two figures illustrate the task.
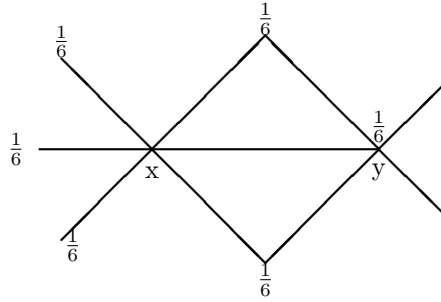
Figure 1. Starting configuration for the transport plan; mass 0 at all vertices without number attached.
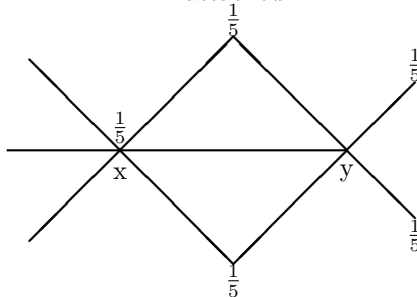
Figure 2. Target configuration for the transport plan

We can also recall the duality formula (2.3) and consider the following 1-Lipschitz function. From this function, we clearly see why triangles, that is common neighbors of the vertices x and y contribute to decreasing the transportation cost.
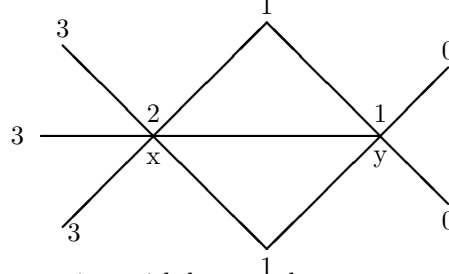
Figure 3. Mass moved from vertices with larger value

In fact, not only triangles, but also quadrangles and pentagons (but not polygons with more edges) influence Ricci curvature.

The lower bound of Theorem 3.1 is sharp both for complete graphs and for trees. On a complete graph $\mathcal{K}_n$ ($n \geq 2$) with n vertices, $\sharp(x,y) = n - 2$ for any x, y. Hence the inequality

$$\kappa(x,y) \geq \frac{n-2}{n-1}$$

is sharp. That trees also attain the lower bound of Theorem 3.1, follows from the fact that on n a tree $T = (V, E)$, for any neighboring x, y,

$$(3.13) \qquad \kappa(x,y) = -2\left(1 - \frac{1}{d_x} - \frac{1}{d_y}\right)_+.$$

We can also relate this to the above heuristic discussion of the relation between Ricci curvature and the relative volume of the intersection of balls. In fact, $\sharp(x,y)/d_x \vee d_y$ is $m_x \wedge m_y(G) := m_x(G) - (m_x - m_y)_+(G)$, i.e. the intersection measure of m_x and m_y. The vertices x_1 that satisfy $x_1 \sim x$, $x_1 \sim y$ constitute the intersection of the unit metric spheres centered at x and y, resp.

We also have an easy upper bound for the Ricci curvature of a graph.

Theorem 3.2. *On a locally finite graph $G = (V, E)$, for any neighboring x, y, we have*

$$(3.14) \qquad \kappa(x,y) \leq \frac{\sharp(x,y)}{d_x \vee d_y}.$$

We now consider the local clustering coefficient of Watts-Strogatz [19]

$$(3.15) \qquad c(x) := \frac{1}{d_x(d_x - 1)} \sum_{y,y \sim x} \sharp(x,y).$$

$c(x)$ measure the extent to which neighbors of x are directly connected. Expressed in words,

$$(3.16) \qquad c(x) = \frac{\text{number of realized edges between neighbors of } x}{\text{number of possible edges between neighbors of } x}.$$

This clustering coefficient is an important quantity in network analysis. For instance, in social networks where the vertices represent individuals and the edges friendship relations, the question addressed by the clustering coefficient is "How many of the friends of my friends are also my friends?".

We may also consider this local clustering coefficient as an average over the $\sharp(x,y)$ for the neighbors of x. As such an average, we should also try to compare it to averaged Ricci curvature. In other words, we should consider the discrete version of scalar curvature,

$$(3.17) \qquad \kappa(x) := \frac{1}{d_x} \sum_{y,y \sim x} \kappa(x,y).$$

73

This scalar curvature $\kappa(x)$ and the local clustering coefficient $c(x)$ then control each other. Indeed, from Theorems 3.1 and 3.2, with $D(x) := \max_{y,y \sim x} d_y$, we have

$$\frac{d_x - 1}{d_x} c(x) \geq \kappa(x) \geq -2 + \frac{d_x - 1}{d_x \vee D(x)} c(x).$$

3.2.1. *Stochastic processes on graphs.* We consider a graph with a lower Ricci bound

$$\kappa(x, y) \geq k \text{ for all } x \sim y, \tag{3.18}$$

or equivalently,

$$W_1(m_x, m_y) \leq (1 - k)d(x, y) = 1 - k \text{ for all } x \sim y. \tag{3.19}$$

We shall now interpret this in probabilistic terms as a path coupling criterion for random walks. This translates a lower bound of the Ollivier-Ricci curvature into a control on the expectation value of the distance between two coupled random walks.

By iteration, one may prove that when (3.18) and hence (3.19) holds, then for any t and any $\bar{x}, \bar{y}$, not necessarily neighbors,

$$W_1(\delta_{\bar{x}} P^t, \delta_{\bar{y}} P^t) \leq (1 - k)^t d(\bar{x}, \bar{y}). \tag{3.20}$$

In order to link this to Ricci curvature, we now consider two random walks $(\bar{X}_t, \bar{Y}_t)$ with distributions $\delta_{\bar{x}} P^t, \delta_{\bar{y}} P^t$ that are coupled in the sense that the joint probabilities satisfy

$$p(\bar{X}_t = \bar{x}', \bar{Y}_t = \bar{y}') = \xi_t^{\bar{x}, \bar{y}}(\bar{x}', \bar{y}'),$$

where $\xi_t^{\bar{x}, \bar{y}}(\cdot, \cdot)$ is the optimal coupling of $\delta_{\bar{x}} P^t$ and $\delta_{\bar{y}} P^t$ as in the definition of the Wasserstein distance W_1. The term $W_1(\delta_{\bar{x}} P^t, \delta_{\bar{y}} P^t)$ then becomes the expectation value of the distance $\mathbf{E}^{\bar{x}, \bar{y}} d(\bar{X}_t, \bar{Y}_t)$ between the coupled random walks $\bar{X}_t$ and $\bar{Y}_t$.

Corollary 3.1. *If (3.18) holds, then for any $\bar{x}, \bar{y} \in V$,*

$$\mathbf{E}^{\bar{x}, \bar{y}} d(\bar{X}_t, \bar{Y}_t) = W_1(\delta_{\bar{x}} P^t, \delta_{\bar{y}} P^t) \leq (1 - k)^t d(\bar{x}, \bar{y}). \tag{3.21}$$

3.2.2. *Weighted and neighborhood graphs.* Following [4], we now translate the properties of random walks into geometric structures, the neighborhood graphs. In Section 3.2.3, we shall then use this construct to derive eigenvalue bounds in terms of lower Ricci curvature bounds on graphs.

For this purpose, we shall need to consider weighted graphs, and also allow for the possibility of self-loops. That is, for any $x, y \in V$, not necessarily different, we have a symmetric, nonnegative connection weight

$$w_{xy} = w_{yx} \geq 0. \tag{3.22}$$

We can then declare x and y to be neighbors, $x \sim y$, iff $w_{xy} > 0$. Of course, the unweighted graphs that we have considered before constitute the special cases where $w_{xy} = 1$ iff $x \sim y$ and $w_{xy} = 0$ else. As mentioned, here, we also allow for the possibility of self-loops, that is, vertices x with $w_{xx} > 0$.

Remark: Of course, one could also allow for non-symmetric or negative weights. The spectrum of non-symmetric graphs was systematically investigated in [2], and some results on graphs with possibly negative connection weights can be found, for instance, in [3, 1]. For our present purposes, however, the class of weighted graphs satisfying (3.22) suffices.

The preceding constructions and results can be extended to weighted graphs. We now define the measure m_x by

$$m_x(y) := \frac{w_{xy}}{d_x}, \text{ where now } d_x := \sum_y w_{xy}. \tag{3.23}$$

We can again consider $m_x(y)$ as the probability that a random walker starting at x moves to y in one time step. Since now possibly $m_x(x) > 0$, because there might be a self-loop at x, the random walker might now be lazy and simply stay at x.

As before , the L^2-product is

$$(u, v) = \sum_x d_x u(x) v(x). \tag{3.24}$$

The Laplacian

$$(3.25) \qquad \Delta v(x) = \frac{1}{d_x} \sum_y w_{xy} v(y) - v(x) = \sum_y m_x(y) v(y) - v(x)$$

is self-adjoint and nonpositive as before. Hence, the eigenvalues are nonnegative real numbers. We also have a version of Theorem 3.1 for weighted graphs, taken from [5].

Theorem 3.3. *On a weighted graph, we have for neighbors x, y*

$$\kappa(x, y) \geq - \left(1 - \frac{w_{xy}}{d_x} - \frac{w_{xy}}{d_y} - \sum_{x_1 \in N_{xy}} \frac{w_{x_1 x}}{d_x} \vee \frac{w_{x_1 y}}{d_y} \right)_+$$
$$- \left(1 - \frac{w_{xy}}{d_x} - \frac{w_{xy}}{d_y} - \sum_{x_1 \in N_{xy}} \frac{w_{x_1 x}}{d_x} \wedge \frac{w_{x_1 y}}{d_y} \right)_+$$
$$+ \sum_{x_1 \in N_{xy}} \frac{w_{x_1 x}}{d_x} \wedge \frac{w_{x_1 y}}{d_y} + \frac{w_{xx}}{d_x} + \frac{w_{yy}}{d_y}.$$

Again, this inequality is sharp.

With the notation

$$\mu P(\cdot) = \sum_x \mu(x) m_x(\cdot),$$

the Dirac measure δ_x at x and $\delta_x P^1(\cdot) = \delta_x P(\cdot) = m_x(\cdot)$, the distribution of a t-step random walk starting at x with transition probability m_x becomes

$$(3.26) \qquad \delta_x P^t(\cdot) = \sum_{x_1, \dots, x_{t-1}} m_x(x_1) m_{x_1}(x_2) \cdots m_{x_{t-1}}(\cdot)$$

for $t > 1$. The probability that the random walker moves from x to y in t steps then is

$$(3.27) \qquad \delta_x P^t(y) = \begin{cases} \sum_{x_1, \dots, x_{t-1}} \frac{w_{xx_1}}{d_x} \frac{w_{x_1 x_2}}{d_{x_1}} \cdots \frac{w_{x_{t-1} y}}{d_{x_{t-1}}}, & \text{if } t > 1; \\ \frac{w_{xy}}{d_x}, & \text{if } t = 1. \end{cases}$$

We now define a family of graphs $G[t]$ for $t \geq 1$ whose weights equal the transition probabilities of the t-step random walks on the graph G.

Definition 3.1. The neighborhood graph $G[t] = (V, E[t])$ of the graph $G = (V, E)$ of order $t \geq 1$ is the weighted graph with vertex set V and edge weights

$$(3.28) \qquad w_{xy}[t] := \delta_x P^t(y) d_x$$

from (3.27).

Obviously, $G = G[1]$. Also, $w_{xy}[t] > 0$ if and only if there exists a path of length t between x and y in G.

We now describe the important properties of the neighborhood graph $G[t]$, its Laplacian $\Delta[t]$ and the eigenvalues $\lambda_i[t]$, see [4, 5].

Lemma 3.1. (i) *t even: $G[t]$ is connected iff G is not bipartite. $G[t]$ is not bipartite.*
(ii) *t odd: $G[t]$ is always connected and $G[t]$ is bipartite iff G is bipartite.*
(iii) *$d_x[t] = d_x$ for all $x \in V$, and the inner product (3.24) is the same on all the $G[t]$.*
(iv) *The Laplacian on $G[t]$ is*

$$(3.29) \qquad \Delta[t] = -\text{id} + (\text{id} + \Delta)^t.$$

(v) *Therefore, for even t, the eigenvalues of $\Delta[t]$ satisfy*

$$(3.30) \qquad 0 = \lambda_0[t] \leq \lambda_1[t] \leq \dots \leq \lambda_{N-1}[t] \leq 1.$$

(The smaller upper bound 1 as compared with the bound 2 discussed above stems here from the self-loops of $G[t]$.)

(vi) *Let $d[t](x,y)$ be the distance on $G[t]$ defined as the smallest number of edges needed for a path connecting x and y (this is independent of the weights, except that vertices ξ and η are connected by an edge iff $w_{\xi\eta} > 0$). Then*

$$(3.31) \qquad \frac{1}{t}d(x,y) \leq d[t](x,y),$$

with the convention $d[t](x,y) = \infty$ if $G[t]$ is not connected and x and y are in different components. Conversely, if $E \subseteq E[t]$, then

$$(3.32) \qquad d[t](x,y) \leq d(x,y).$$

In [4], the relationship between the eigenvalues of the original graph G and those of its neighborhood graphs was analyzed.

Proposition 3.1. (i) *If $\lambda_1[t] \geq \mathcal{A}[t]$, then*

$$(3.33) \qquad 1 - (1 - \mathcal{A}[t])^{\frac{1}{t}} \leq \lambda_1 \leq \cdots \leq \lambda_{N-1} \leq 1 + (1 - \mathcal{A}[t])^{\frac{1}{t}}$$

if t is even and

$$(3.34) \qquad 1 - (1 - \mathcal{A}[t])^{\frac{1}{t}} \leq \lambda_1$$

if t is odd.

(ii) *If $\lambda_{N-1}[t] \leq \mathcal{B}[t]$, then all eigenvalues of Δ are contained in*

$$\left[0, 1 - (1 - \mathcal{B}[t])^{\frac{1}{t}}\right] \bigcup \left[1 + (1 - \mathcal{B}[t])^{\frac{1}{t}}, 2\right]$$

for even t, whereas

$$\lambda_{N-1} \leq 1 - (1 - \mathcal{B}[t])^{\frac{1}{t}}$$

for odd t.

Thus, eigenvalues bounds on $G[t]$ translate into eigenvalue bounds on the original graph G. This is a powerful principle for estimating the eigenvalues of G. As the neighborhood graphs constitute a geometric representation of the random walk on G, this can be seen as a scheme for translating properties of the random walk into eigenvalue bounds.

3.2.3. *Ricci curvature and eigenvalues of graphs.* In this section, we assume that the graph G is finite, that is, it has finitely many, say N, vertices, and then also finitely many edges. Here, we follow [5] to estimate the eigenvalues in terms of the Ricci curvature. Ollivier [16] showed

Theorem 3.4. *When we have a lower Ricci curvature bound*

$$(3.35) \qquad \kappa(x,y) \geq k,$$

(in fact, it suffices to have this for all $x \sim y$), then

$$(3.36) \qquad k \leq \lambda_1 \leq \ldots \leq \lambda_{N-1} \leq 2 - k.$$

A problem with this estimate is that for most graphs, $k \leq 0$ in (3.35), so that (3.36) only yields a trivial estimate. We shall therefore develop an estimate of [5] which is nontrivial for all connected finite graphs that are not bipartite.

Lemma 3.2. *Let k be a lower bound of κ on G. If $E \subseteq E[t]$, then the curvature $\kappa[t]$ of the neighborhood graph $G[t]$ satisfies*

$$(3.37) \qquad \kappa[t](x,y) \geq 1 - t(1-k)^t, \quad \forall x,y \in V.$$

We can now see the upper bound of the largest eigenvalue in Theorem 3.4. W.l.o.g. $k > 0$, in which case $E \subset E[t]$. From Lemma 3.2 and $\lambda_1 \geq k$, we know on $G[t]$,

$$\lambda_1[t] \geq 1 - t(1-k)^t.$$

Then with Proposition 3.1 (i), for even t,

$$\lambda_{N-1} \leq 1 + t^{\frac{1}{t}}(1-k).$$

Letting $t \to +\infty$ yields $\lambda_{N-1} \leq 2 - k$, indeed.

The neighborhood graph technique then leads to the following generalization of Theorem 3.4, the main result of [5].

Theorem 3.5. *Let $k[t]$ be a lower bound of Ollivier-Ricci curvature of the neighborhood graph $G[t]$. Then for all $t \geq 1$ the eigenvalues of Δ on G satisfy*

$$(3.38) \qquad 1 - (1 - k[t])^{\frac{1}{t}} \leq \lambda_1 \leq \cdots \leq \lambda_{N-1} \leq 1 + (1 - k[t])^{\frac{1}{t}}.$$

If G is not bipartite, then for all sufficiently large t, $k[t] > 0$, and hence (3.38) is nontrivial in the sense that the lower bound is positive and the upper bound is < 2.

3.3. **Other curvature notions for graphs.** We conclude this brief survey with some curvature notions for graphs other than Ricci curvature.

First, combinatorial curvature: we fill faces into the graph. We therefore assume that the (possibly infinite) graph G is embedded into a 2-manifold $S(G)$ such that each face is homeomorphic to a closed disk with finite edges as the boundary. For instance, G could be a planar graph, that is, a graph embedded into the plane. Therefore, we call such a $G = (V, E, F)$ that can be embedded into a 2-manifold a semiplanar graph. For each vertex $x \in V$, the combinatorial curvature at x is

$$(3.39) \qquad \Phi(x) = 1 - \frac{d_x}{2} + \sum_{\sigma \ni x} \frac{1}{\deg(\sigma)},$$

where, as before, d_x is the degree of the vertex x, whereas $\deg(\sigma)$ is the degree of the face σ. The sum is taken over all faces incident to x (i.e. $x \in \sigma$).

When we replace each face of G with a regular polygon of side lengths one and glue them along the common edges and equip the polygonal surface $S(G)$ with the resulting metric structure, then (3.39) simply measures the difference of 2π and the total angle Σ_x at the vertex x,

$$(3.40) \qquad 2\pi \Phi(x) = 2\pi - \Sigma_x.$$

Let $\chi(S(G))$ denote the Euler characteristic of the surface $S(G)$. We then have the Gauss-Bonnet formula of G of [9],

$$(3.41) \qquad \sum_{x \in G} \Phi(x) \leq \chi(S(G)),$$

whenever $\Sigma_{x \in G : \Phi(x) < 0} \Phi(x)$ converges. Thus, the combinatorial curvature captures a topological property of semiplanar graphs.

We can also compare the combinatorial curvature with another version of curvature naturally obtained from the surface $S(G)$, its generalized sectional (Gaussian) curvature. It turns out that the semiplanar graph G has nonnegative combinatorial curvature precisely if the polygonal surface S(G) is an Alexandrov space with nonnegative sectional curvature, i.e. Sec $S(G) \geq 0$ (or Sec$(G) \geq 0$ for short). This principle is systematically explored in [11].

A metric space (X, d) on which each pair of points in X can be joined by a shortest path is called an Alexandrov space if locally satisfies the Toponogov triangle comparison. Essentially, nonnegative curvature in the present context means that the total angles of geodesic triangles are at least 2π. Monographs on Alexandrov spaces are [7, 6].

REFERENCES

[1] F. Bauer, F. M. Atay, and J. Jost. Synchronized chaos in networks of simple units. *EPL (Europhysics Letters)*, 89(2):20002, 2010.

[2] Frank Bauer. Normalized graph Laplacians for directed graphs. *Linear Algebra Appl.*, 436(11):4193–4222, 2012.

[3] Frank Bauer, Fatihcan M. Atay, and Jürgen Jost. Synchronization in discrete-time networks with general pairwise coupling. *Nonlinearity*, 22(10):2333–2351, 2009.

[4] Frank Bauer and Jürgen Jost. Bipartite and neighborhood graphs and the spectrum of the normalized graph Laplace operator. *Commun. Anal. Geom.*, 21(4):787–845, 2013.

[5] Frank Bauer, Jürgen Jost, and Shiping Liu. Ollivier-Ricci curvature and the spectrum of the normalized graph Laplace operator. *Mathematical research letters*, 19(6):1185–1205, 2012.

[6] V.N. Berestovskij and I.G. Nikolaev. Multidimensional generalized Riemannian spaces. In *Geometry IV. Nonregular Riemannian geometry. Transl. from the Russian by E. Primrose.* Berlin: Springer-Verlag, 1992.

[7] D. Burago, Yu. Burago, and S. Ivanov. *A course in metric geometry.* Providence, RI: American Mathematical Society (AMS), 2001.

[8] Fan R.K. Chung. *Spectral graph theory.* Providence, RI: AMS, American Mathematical Society, 1997.

[9] Matt DeVos and Bojan Mohar. An analogue of the Descartes-Euler formula for infinite graphs and Higuchi's conjecture. *Transactions of the American Mathematical Society*, 359(7):3287–3300 (electronic), 2007.

[10] C. Fr. Gauß. Allgemeine Flächentheorie. (Disquisitiones generales circa superficies curvas.) (1827.). Deutsch herausgeg. von A. Wangerin. 5. Aufl. Leipzig: Akad. Verlagsges., 64 S. 8° (1921). (Ostwalds Klassiker der exakten Wissenschaften, Nr. 5.) (1921)., 1921.

[11] Bobo Hua, Jürgen Jost, and Shiping Liu. Geometric analysis aspects of infinite semiplanar graphs with non-negative curvature. *arXiv.org*, July 2011.

[12] Jürgen Jost. *Riemannian geometry and geometric analysis. 6th ed.* Berlin: Springer, 6th ed. edition, 2011.

[13] Jürgen Jost. *Mathematical methods in biology and neurobiology.* London: Springer, 2014.

[14] Jürgen Jost and Shiping Liu. Ollivier's Ricci curvature, local clustering and curvature dimension inequalities on graphs. *Discrete and Computational Geometry*, 51(2):300–322, March 2014.

[15] Yong Lin and Shing-Tung Yau. Ricci curvature and eigenvalue estimate on locally finite graphs. *Mathematical research letters*, 17(2):343–356, 2010.

[16] Yann Ollivier. Ricci curvature of Markov chains on metric spaces. *Journal of Functional Analysis*, 256(3):810–864, 2009.

[17] Yann Ollivier. A survey of Ricci curvature for metric spaces and Markov chains. In *Probabilistic approach to geometry. Proceedings of the 1st international conference, Kyoto , Japan, 28th July – 8th August, 2008*, pages 343–381. Tokyo: Mathematical Society of Japan (MSJ), 2010.

[18] Bernhard Riemann. *Bernhard Riemann "Über die Hypothesen, welche der Geometrie zu Grunde liegen". Historisch und mathematisch kommentiert von Jürgen Jost.* Berlin: Springer Spektrum, 2013.

[19] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world' networks. *nature*, 393(6684):440–442, 1998.

Robust Shape Reconstruction
and Optimal Transportation

Pierre Alliez, Simon Giraudot, and David Cohen-Steiner

Abstract

We describe a framework for robust shape reconstruction from raw point sets, based on optimal transportation between measures, where the input point sets are seen as distribution of masses. In addition to robustness to defect-laden point sets, hampered with noise and outliers, our approach can reconstruct smooth closed shapes as well as piecewise smooth shapes with boundaries.

1. Introduction

Assuming a geometric data set made out of points or slices, the process of shape reconstruction amounts to recovering a surface or a solid that matches these samples. This problem is inherently ill-posed as infinitely-many shapes may fit the data. One must thus regularize the problem and add priors such as simplicity or smoothness of the inferred shape. In addition, the increasing variety of sensors for acquiring point sets corresponds to a range of defects inherent to each sensor and associated acquisition process. The point sets may differ in terms of sampling, noise and outliers. In addition, the level of noise may vary within the same point set, depending on the type of noise, acquisition condition and light-material interaction. Our quest for robustness includes the ability to deal with variable noise.

Related Work. In past years the *smooth, closed case* (i.e., shapes without sharp features nor boundaries) has received considerable attention. Computational geometric approaches to surface reconstruction, commonly based on Delaunay triangulations, generally provide theoretical guarantees under specific sampling models [4, 7]. The search for increased robustness to noise led to a wide variety of methods involving denoising, integral computations, variational formulations or scale-space processes. Robustness to outliers has been investigated through outlier removal [16], data clustering [15], robust norms such as the $l1$-sparse norm [2], spectral methods [12], or robust distances. However, state-of-the-art methods have several shortcomings: in addition to being generally not robust to outliers and not sufficiently robust to noise, they often require additional input attributes, such as lines of sight or oriented normal vectors [11].

Moving from the smooth, closed case to the *piecewise smooth case* (possibly with boundaries) is considerably harder as the ill-posedness of the problem applies to each sub-feature of the inferred shape. Further, very few approaches tackle the combined issue of robustness (to sampling defects, noise and outliers) and feature reconstruction [8, 1], and none to our knowledge addresses all the issues from heterogeneous data.

Positioning. In this work we develop shape reconstruction methods that are robust under perturbations of the data in the sense of the optimal transportation distance, the input point samples being considered as Dirac measures. Optimal transportation refers to the problem of optimizing

the cost of transportation of resources [17]. An intuitive example of optimal transport consists in determining the most effective way to move a pile of sand to a hole of the same volume. Most effective herein means that the integral of the distances the sand is moved, one infinitesimal unit of volume at a time, is minimal. This formulation of the problem, referred to as Monge's variational formulation, assumes that the sand is moved through a point-to-point mapping denoted by the transport plan. This restriction was relaxed by Kantorovich who extended the formulation to deal with transport plans between two probability measures. Optimal transportation reveals a versatile framework for geometry processing when observing that it can robustly measure distances between surfaces [13], and more generally between measures, be they discrete or continuous [5]. In addition to being symmetric, this distortion measure is by construction noise and outlier robust, a highly desirable feature when seeking robustness to defect-laden data. It has also been used for surface comparison [13] and displacement interpolation [3].

2. Smooth Reconstruction

We first focus on shapes that are both smooth and closed (with no boundary). We describe a recent noise-adaptive robust distance function which relies on the only assumption that the inferred shape is a smooth submanifold of known dimension [9]. Our algorithm takes as input a raw point set sampling the boundary of the inferred solid object.

Chazal et al. [5] introduced a robust distance function from a query point x to a measure μ in $\mathbb{R}$:

$$d_{\mu,m}^2 : \mathbb{R}^n \to \mathbb{R}, \ x \mapsto \frac{1}{m} \int_{B(x, r_{\mu,m}(x))} \|x - y\|^2 d\mu(y),$$

where $m \in (0; 1]$ denotes a user-defined mass and $r_{\mu,m}$ denotes the minimal radius such that the ball $B(x, r_{\mu,m}(x))$ encloses this mass m. Considering the input point set as a distribution of n point masses, this simplifies to a sum on the K nearest neighbors.

In our quest for robustness, this function exhibits two relevant properties that are its robustness in the Wasserstein distance, and the 1-semiconcavity of its square. In addition, it is shown that under suitable sampling conditions, the sublevel sets of this distance function provide a homotopically correct approximation of the inferred shape, even in the presence of noise and outliers.

Mullen et al. [14] used these properties of the distance function to surface reconstruction with robustness to noise, outliers and missing data. A major limitation of this approach comes from the mass parameter K which must be user-specified: there is no principled way to select it automatically and it requires a trial-error process to trade robustness for accuracy. In addition, K is a global parameter and hence can not deal with non-uniform noise. The first two curves of Figure 2.3 show the behavior of the function on a point set with variable noise.

We now define a noise-adaptive distance function that solely relies on a dimension assumption: the inferred shape is a smooth submanifold of known dimension k embedded in a space of dimension d. For an input measure μ and a constant parameter $\alpha > 0$, we define the adaptive distance function as follows:

$$\delta_{\mu,\alpha} = \inf_{m>0} \frac{d_{\mu,m}}{m^\alpha}.$$

First, consider the case of an ambient noise μ in dimension d as shown on Figure 2.1 (left). The distance function simplifies to $d_{\mu,m}^2 = c_1 m^{2/d}$, with c_1 a constant depending on k and on the density of μ. For $\alpha > \frac{1}{d}$, we thus get that $d_{\mu,m}/m^\alpha$ is decreasing with m. Therefore its minimum is reached at a value $m^* = 1$, and $\delta_{\mu,\alpha}^2$ is, in the discrete case, the average squared distance to the whole point set.

Consider now a uniform continuous measure μ on a k-subspace in d-dimensional space. At distance h from this subspace, we find that

$$d_{\mu,m}^2(h) = c_1 m^{2/k} + h^2.$$

Hence for $\alpha < \frac{1}{k}$, $d_{\mu,m}(h)/m^\alpha$ is unimodal as a function of m, and $m^*(h) = c_2 h^k$. We obtain: $\delta_{\mu,\alpha}(h) = c_3 h^{1/k-\alpha}$. Notice that this function reaches zero on the shape and grows with a vertical tangent as we move away from it: its level sets therefore accumulate way tighter around the data than the non-adaptive distance function which is a smooth quadratic function that never goes to zero. We thus obtain a precise localization of the inferred shape.
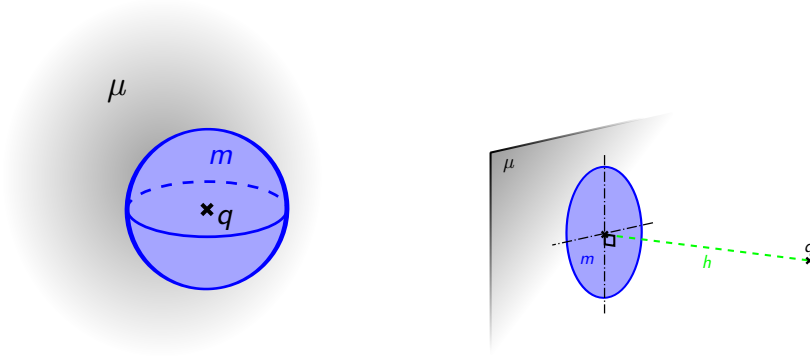
Figure 2.1: **Left: Ambient noise μ in a d-dimensional space.** A sphere centered on q contains the mass m. **Right: Uniform measure μ on a k-subspace.** A circle centered on the orthogonal projection of q on the supporting plane of μ contains the mass m.

Figure 2.2 depicts a realistic case of a noisy k-submanifold on a d-dimensional space. Depending on the value of m, it boils down to one of the two previous cases. Consider a query point q lying on the shape. At smaller scale than the noise level, the data appears as an ambient noise: for $\alpha > \frac{1}{d}$, on small values of m, $d_{\mu,m}(x)/m^\alpha$ is decreasing. As soon as the scale gets larger than the noise level, the data appears as a k-submanifold: for $\alpha < \frac{1}{k}$, $d_{\mu,m}(x)/m^\alpha$ is increasing.

Scale $m = 10$ nearest neighbors **Scale $m = 30$ nearest neighbors**
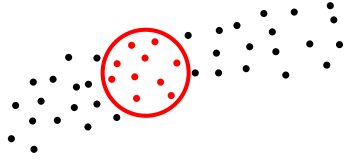
- Apparent dimension $= 2$
- Ambient noise in a 2D space
- $\frac{d_{\mu,m}(q)}{m^\alpha}$ is decreasing for $\alpha > \frac{1}{2}$
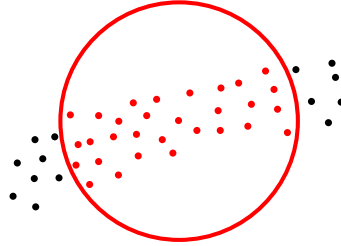
- Apparent dimension $= 1$
- 1-submanifold in a 2D space
- $\frac{d_{\mu,m}(q)}{m^\alpha}$ is increasing for $\alpha < 1$
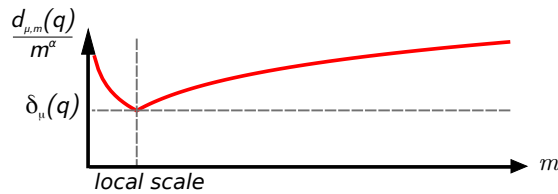
Figure 2.2: **Adaptive distance function.**

This means that under these conditions, the minimum m^* reached relates to the scale of the local noise. Consequently, the adaptive distance function provides an accurate representation of the data on noise-free areas, while sufficiently smoothing the data in poorly sampled areas. These properties are met as long as $\alpha \in]\frac{1}{d}; \frac{1}{k}[$. We thus choose $\alpha = \frac{3}{4}$ for curves in 2D and $\alpha = \frac{5}{12}$ for surfaces in 3D.

This parameter can also be set to deal with a curve in 3D: $\alpha = \frac{2}{3}$. However, this is out of scope in our context as we aim at reconstructing closed surfaces bounding a 3D solid.

To achieve correct topological inference similarly to the non-adaptive distance function, it is necessary to limit the infimum over the values of m that exceed a threshold m_0. Indeed, for a single value of m, the robust distance $d_{\mu,m}$ is $\frac{1}{\sqrt{m}}$-robust: this means that two measures μ_1 and μ_2

that are ε away in the Wasserstein 2-distance will have robust distances $d_{\mu_1,m}$ and $d_{\mu_2,m}$ at most $\frac{\varepsilon}{\sqrt{m}}$ away in the sup norm. For a single value of m, the robust distance is also 1-semiconcave. All functions in the infimum are therefore $m_0^{-\alpha-1/2}$-robust with $m_0^{-2\alpha}$-semiconcave squares. Defining this non-zero infimum m_0 for the adaptive distance function, we preserve these properties of the original non-adaptive function. In practice, we set a lower bound $K_0 = 6$ for selecting the K nearest neighbors.
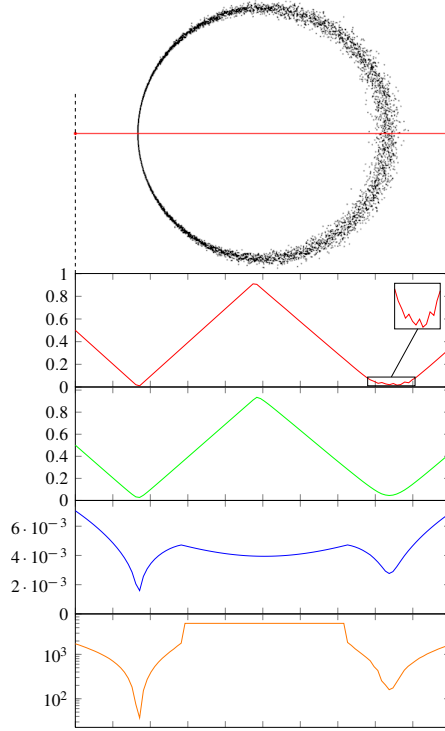
Figure 2.3: **Distance functions. Top:** input point set and segment selected to depict function values. **Red curve:** robust function d_μ with $K = 6$: small details are captured in noise-free area, but the function is noisy on noisy area. **Green curve:** robust function d_μ with $K = 70$: noisy areas are captured, but noise-free areas are over-smoothed and the first minimum of the function is shifted to the right. **Blue curve:** adaptive function δ_μ: all features are captured. **Orange curve:** selected value for K: notice the high dynamic of the function (log vertical scale). The flat maximum appears when the total number of points is reached.

Figure 2.3 compares the adaptive and non-adaptive distance functions. This new function altogether removes the need for a scale parameter and does not exhibit the defects of the globally-scaled and non-adaptive function. A curve also depicts the local scale as value of K where the minimum is reached.

The reconstruction algorithm consists in computing an implicit function whose 0-isolevel defines the shape. It comprises two main steps: (i) Computing the adaptive distance function, and (ii) Signing it by estimating the sign of a signed version of this distance function on a set of discrete locations, and propagating the sign guesses using the adaptive distance function.

Unsigned Distance. We represent the adaptive distance function by linear interpolation on the triangles of an isotropic triangulation. The latter is obtained through adaptive Delaunay refinement such that the interpolated and real values are ε away from each other (see Figure 2.4). For each vertex of the triangulation computing the adaptive distance function amounts to search for a minimum over all possible K nearest neighbors, leading to a major scalability issue. Instead of performing an exhaustive search, we rely on a k-d tree structure for neighbor search, combined to a hierarchical clustering of the input point set [9].
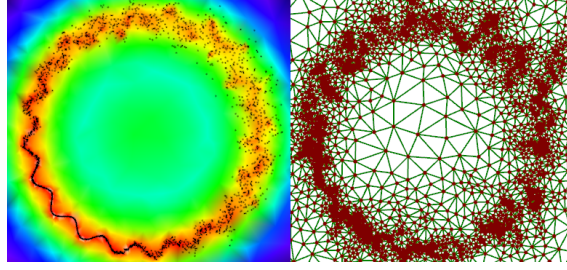
Figure 2.4: **Delaunay refinement.** Left: input point set and distance function. Right: output triangulation.

Signing. We first guess the sign of the output implicit function on a discrete set of points. The latter are selected as the corners of a regular grid that covers the domain of the input set (defined as a loose bounding box). Edges are randomly picked on this grid and each of them is assigned a sign estimate $\varepsilon_{i,j}$ that is -1 or $+1$, depending if the two end nodes i and j are estimated to be on the same side or on opposite sides of the shape. Then, by minimizing the following energy: $E_G(f) = \sum_{(i,j) \in G} (f_i + \varepsilon_{i,j} f_j)^2$, a sign f_i is estimated for all the nodes i. A confidence is derived for each node as the ratio of sign hypothesis on edges that are matched by the output signs of the nodes. Figure 2.5 illustrates the sign guess pipeline.
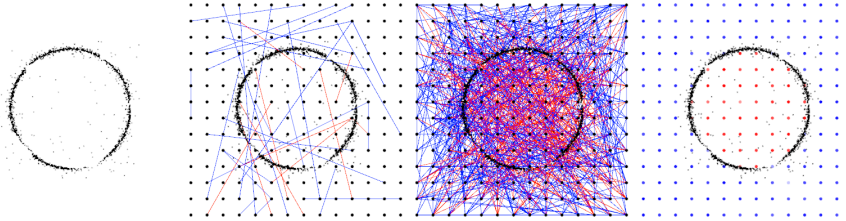
Figure 2.5: **Random graph.** From left to right: input point set; edges of the graph (only 1% of edges are shown for clarity, with blue for similar signs and red for different signs); 20% of the graph edges shown; signed function at graph nodes after linear solve (red for inside, blue for outside).

Given the set of most confident sign guesses and the adaptive unsigned distance function, the output signed implicit function is computed using a method inspired by the random walker approach used for image segmentation [10]. Figure 2.6 illustrates resilience to gradually variable noise: the reconstructed surface smoothly approximates the inferred shape on noisy area while providing high accuracy on noise-free area. We compare to the Poisson reconstruction approach which relies upon a single scale parameter. Both algorithms are timed on GNU/Linux with a 4-core 2.4 GHz *Intel Core i7* processor with 8 GB of RAM.

Figure 2.7 depicts our algorithm at work on a point set generated by photogrammetry. To evaluate resilience to variable noise we add a synthetic noise on the top half part of the point set.

3. Piecewise Smooth Reconstruction

For piecewise smooth shapes with boundaries we reformulated 2D shape reconstruction and simplification in a preliminary work [6] as a transportation problem between measures (*i.e.*, mass distributions), where the input point samples are considered as Dirac measures and the reconstructed shape is seen as the support of a continuous measure defined on the vertices and edges of a 2D triangulation. Our formulation significantly differs from the common way to state the optimal transportation problem in the sense that we do not know the target measure but instead need to solve for it. We restrict the target measure to be defined as the sum of piecewise uniform measures on the simplices of a 2D triangulation. When the transport plan is restricted to transport each input point sample to the nearest edge of the triangulation, the optimal transport cost based on the Wasserstein-2 distance is computed in closed form. The final triangulation is
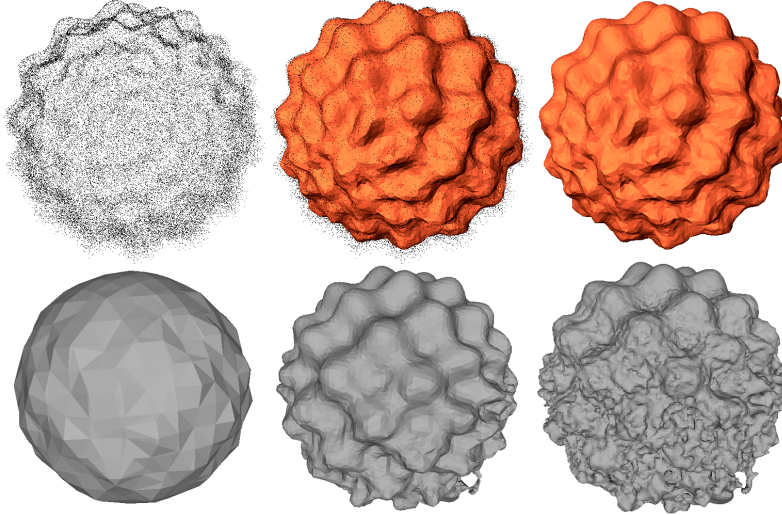
Figure 2.6: **Gradually variable noise (generated).** Top: raw point set, where noise increases linearly from top to bottom; point set & our reconstruction; our reconstruction only (running time: 82s). Bottom: Poisson reconstruction with a uniform octree depth of 4, 6 and 8 (running times: 1, 10 and 50s respectively).
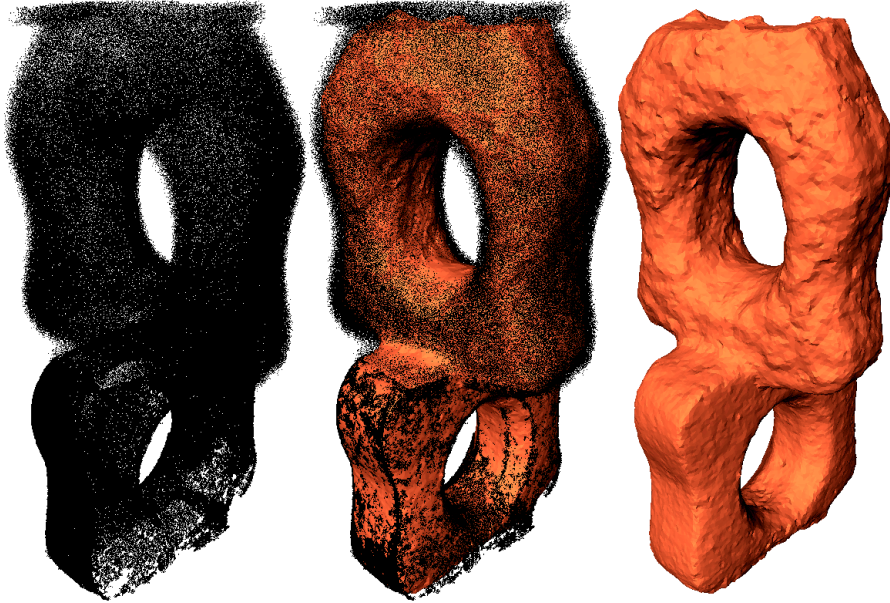
Figure 2.7: **Two levels of noise**. Left: raw point set with additional noise on the top half part. Middle: point set & reconstruction. Right: reconstruction only (running time: 242s).

obtained through greedy decimation of an initial dense Delaunay triangulation constructed with the input point samples, where the decimation operators are ordered so as to minimally increase the total transport cost. This approach both reconstructs and generates a simplified shape, and brings forth a unified treatment of noise, outliers and boundaries (Figure 3.1).

Without restricting the transport plan such that each sample point can be transported to an arbitrary edge of the triangulation there is no closed form anymore as the structure of the transport plan becomes a variable of the problem. As first direction we performed a pointwise discretization of each edge of the triangulation before resorting to a dense linear program to determine the
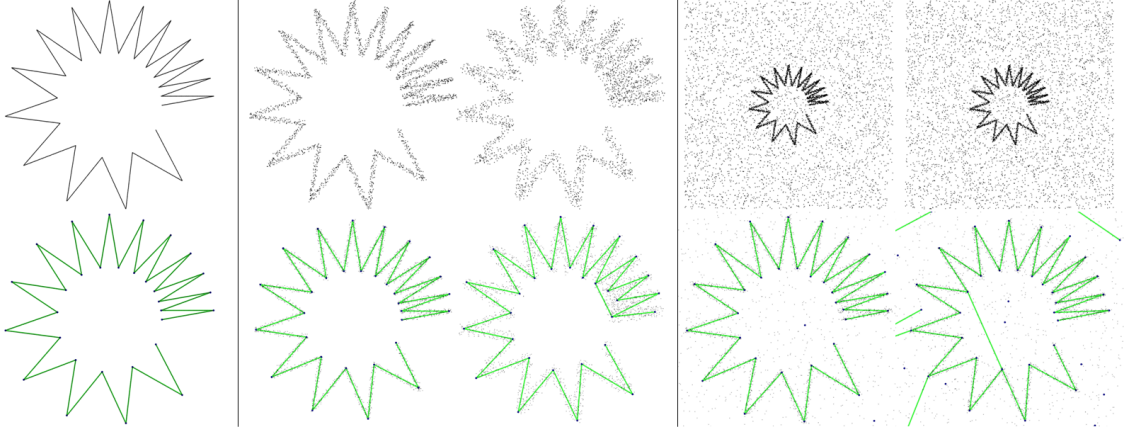
Figure 3.1: **Robust 2D shape reconstruction and simplification.** The input shape (3K points) has sharp corners subtending small angles as well as boundaries. The reconstruction is perfect for a noise-free input (left); as noise is added (middle, 2% and 2.5% of bounding box), the output degrades gracefully, still capturing most of the sharp angles; after adding 4K or 4.5K outliers and 2% of noise (right), the reconstruction remains satisfactory, although artifacts start appearing in this regime.

point-to-point optimal transport plan and associated cost. The latter approach is very computer-intensive but exhibits even better resilience to noise and outliers.

The linear programming approach described above applies to noise and outlier-robust reconstruction of surfaces with sharp features and boundaries. However, and even when using the restricted transport plan as mentioned above, there is no closed form when dealing with the facets of a triangulated surface. Computing the optimal transport cost and plan through pointwise discretization and linear programming is thus substantially more computer-intensive than for 2D shapes.

For surfaces we formulate the reconstruction problem as a transportation problem between measures (i.e., mass distributions), where the input point samples are considered as Dirac measures and the reconstructed shape is seen as the support of a piecewise uniform measure over the simplices (vertices, edges, facets) of a surface simplicial complex. Using three different types of simplices provides us with a means to deduce the local dimension from the optimal transport plan found by the solver.

To obtain high robustness to noise and outliers we formulate the problem with a general transport plan where each input sample point can be split into sub-masses, each transported to different locations on the complex. We thus use a discretized formulation of the optimal transport problem, where we approximate the optimal transport cost between the input point set $\mathcal{S}$ and the simplicial complex $\mathcal{C}$ using quadrature intervals, which we call hereafter *bins*. To this end, we partition vertices, edges, and facets of $\mathcal{C}$ into a set of bins $\mathcal{B}$ and evaluate the optimal cost between $\mathcal{S}$ and $\mathcal{B}$ as the sum of squared distances between the points in $\mathcal{S}$ and the centroid of the bins in $\mathcal{B}$. Bins on facets are constructed as the bounded Voronoi cells of a centroidal Voronoi tessellation (see inset). In order to accommodate the non-uniform distribution of bins, we assign a *capacity* for each bin in $\mathcal{B}$ (i.e., the total amount of mass that a bin can receive). While we set vertex bins to unit capacity, edge bins and facet bins are respectively assigned capacities proportional to the lengths and areas of the Voronoi cells.
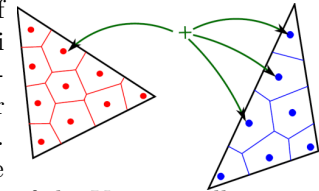
We use a linear programming formulation to compute the optimal transport cost between the input point set $\mathcal{S}$ and the bin set $\mathcal{B}$. In the following, we denote the simplices of $\mathcal{C}$ as $\{\sigma_j\}_{j=1\cdots L}$ and the bins in $\mathcal{B}$ as $\{b_j\}_{j=1\cdots M}$, where L and M are the number of simplices and bins respectively. We also define $s(j)$ to be the index of the simplex containing the bin b_j (i.e., $b_j \in \sigma_{s(j)}$). The capacity c_j of bin b_j is defined as the ratio between the bin's area and the area of its containing

simplex. Finally, we denote m_{ij} as the amount of mass transported from point $p_i \in S$ to the centroid of bin b_j.

With these definitions, we can now refer to a *transport plan* between $\mathcal{S}$ and $\mathcal{B}$ as a set of $N \times M$ variables m_{ij} such that:

$$(3.1) \qquad\qquad\qquad \forall ij : \; m_{ij} \geq 0,$$

$$(3.2) \qquad\qquad\qquad \forall i : \; \sum_j m_{ij} = m_i,$$

$$(3.3) \qquad\qquad \forall j_1, j_2 \text{ s.t. } s(j_1) = s(j_2) : \; \frac{1}{c_{j_1}} \sum_i m_{ij_1} = \frac{1}{c_{j_2}} \sum_i m_{ij_2},$$

where Equation 3.2 ensures that the entire measure of a point gets transported onto simplices, and Equation 3.3 ensures a uniform measure over each simplex of $\mathcal{C}$. An *optimal transport plan* is then defined as a transport plan π that minimizes the associated transport cost

$$cost(\pi) = \sum_{ij} m_{ij} \|p_i - b_j\|^2.$$

Finding a transport plan minimizing the transport cost results in a linear program with respect to the m_{ij}, with equality (Eq. 3.2 and 3.3) as well as inequality constraints (Eq. 3.1). In order to enforce the uniformity constraint (Eq. 3.3), we also introduce L additional variables l_i (one per simplex σ_i) indicating the target measure density of the corresponding simplex. The final formulation is thus:

$$\text{Minimize} \quad \sum_{ij} m_{ij} \|p_i - b_j\|^2$$

w.r.t. the variables m_{ij} and l_j, and subject to:

$$\begin{cases} \forall i : \displaystyle\sum_j m_{ij} = m_i \\[2mm] \forall j : \displaystyle\sum_i m_{ij} = c_j \cdot l_{s(j)} \\[2mm] \forall i,\, j : m_{ij} \geq 0,\, l_j \geq 0 \end{cases}$$

Our reconstruction algorithm proceeds by iterative decimation – through half edge collapse operators – of a 2-simplicial complex initialized as the facets of the 3D Delaunay triangulation constructed from a subset of the input points (Figure 3.2). We also couple our decimation with an optimization procedure in order to relocate the vertices in the reconstructed simplicial complex. This optimization procedure further minimizes the transport cost through alternating vertex relocation (with a fixed transport plan) and re-computation of the optimal transport plan. The edge collapse operators are our means to generate a simplicial complex with anisotropic triangle facets where relevant from the approximation point of view.

4. Conclusion

We described two recent contributions for noise and outlier robust shape reconstruction, both derived from computing and minimizing distances between measures through optimal transportation.

For smooth, closed shapes we compute and sign a noise-adaptive distance function before extracting one its isolevel as final reconstruction. For piecewise smooth shapes with boundaries we proceed by iterative, feature-preserving simplification of a simplicial complex constructed from the input point set. To achieve noise and outlier robustness, an error metric driving the simplification is derived in terms of optimal transport between the input point set and the reconstructed mesh, both seen as mass distributions. Even when restricting the transport plan to transport input samples to local neighborhoods, our solution based on linear programming has a high computational cost: we need to gain 4 orders of magnitude to obtain a practical algorithm. In the future we plan to investigate how to compute an estimate of the optimal transport cost. We also plan to perform a statistical analysis of the optimal transport plan per simplex in order to devise a more parsimonious shape refinement algorithm. Intuitively, the transport cost will be used to decide
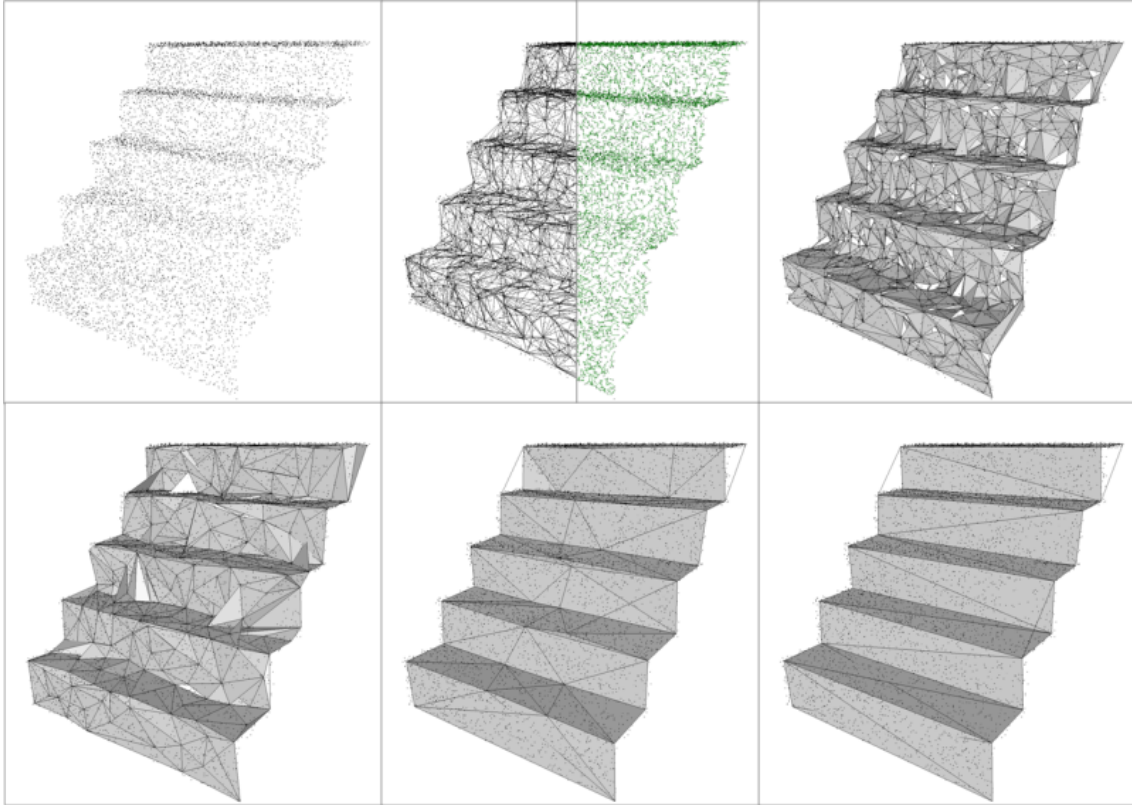
Figure 3.2: **Robust surface reconstruction and simplification.** Top: Initial point set; Filtered 3D Delaunay triangulation of a random subset containing 10% of the input points and initial transport plan assigning point samples to discretization points (green arrows); and first few decimation step. Bottom: Reconstruction with 100, 50, and 14 vertices, respectively.

where to refine the shape, while the transport plan analysis translates into how to refine, namely into the type and parameters of a local anisotropic refinement operator.

REFERENCES

[1] Haim Avron, Andrei Sharf, Chen Greif, and Daniel Cohen-Or. ℓ_1-sparse reconstruction of sharp point set surfaces. *ACM Trans. on Graphics*, 29(5):1–12, 2010.

[2] Haim Avron, Andrei Sharf, Chen Greif, and Daniel Cohen-Or. L1-sparse reconstruction of sharp point set surfaces. *ACM Transactions on Graphics*, 29:135:1–135:12, November 2010.

[3] Nicolas Bonneel, Michiel van de Panne, Sylvain Paris, and Wolfgang Heidrich. Displacement interpolation using lagrangian mass transport. *ACM Transactions on Graphics (SIGGRAPH Asia Proceedings)*, 2011.

[4] Frédéric Cazals and Joachim Giesen. Delaunay triangulation based surface reconstruction. In J.D. Boissonnat and M. Teillaud, editors, *Effective Computational Geometry for Curves and Surfaces*, pages 231–276. Springer-Verlag, Math. and Visualization, 2006.

[5] Frédéric Chazal, David Cohen-Steiner, and Quentin Mérigot. Geometric inference for probability measures. *Journal on Foundations of Computational Mathematics*, 11:733–751, 2011.

[6] Fernando de Goes, David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. An Optimal Transport Approach to Robust Reconstruction and Simplification of 2D Shapes. *Computer Graphics Forum*, 30(5):1593–1602, 2011. Special issue for EUROGRAPHICS Symposium on Geometry Processing.

[7] Tamal K. Dey. *Curve and Surface Reconstruction: Algorithms with Mathematical Analysis*. Cambridge Monographs on Applied and Computational Mathematics, 2006.

[8] S. Fleishman, D. Cohen-Or, and C.T. Silva. Robust moving least-squares fitting with sharp features. In *ACM SIGGRAPH 2005 Papers*, page 552, 2005.

[9] Simon Giraudot, David Cohen-Steiner, and Pierre Alliez. Noise-Adaptive Shape Reconstruction from Raw Point Sets. *Computer Graphics Forum*, 32(5):229–238, 2013.

[10] Leo Grady. Random walks for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11):1768–1783, 2006.

[11] Michael Kazhdan, M. Bolitho, and Hugues Hoppe. Poisson Surface Reconstruction. In *Symposium on Geometry Processing*, pages 61–70, 2006.

[12] Ravikrishna Kolluri, Jonathan Shewchuk, and James O'Brien. Spectral surface reconstruction from noisy point clouds. In *Proceedings of EUROGRAPHICS Symposium on Geometry Processing*, pages 11–21, 2004.

[13] Yaron Lipman and Ingrid Daubechies. Surface comparison with mass transportation. ArXiv preprint 0912.3488, 2010.

[14] Patrick Mullen, Fernando De Goes, Mathieu Desbrun, David Cohen-Steiner, and Pierre Alliez. Signing the unsigned: Robust surface reconstruction from raw pointsets. *Computer Graphics Forum*, 29(5):1733–1741, 2010. Proceedings of EUROGRAPHICS Symposium on Geometry Processing.

[15] Yuqing Song. Boundary fitting for 2D curve reconstruction. *The Visual Computer*, 26:187–204, 2010.

[16] S. Sotoodeh. Outlier detection in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(5):297–302, 2006.

[17] C. Villani. *Topics in Optimal Transportation*. American Mathematical Society, 2010.

Inria Sophia Antipolis - Mediterranée, 2004 route des Lucioles, 06902 Sophia Antipolis, FRANCE • pierre.alliez@inria.fr • simon.giraudot@inria.fr • david.cohen-steiner@inria.fr

The Gromov-Hausdorff distance: a brief tutorial on some of its quantitative aspects

Facundo Mémoli

Abstract

We recall the construction of the Gromov-Hausdorff distance. We concentrate on quantitative aspects of the definition and on quantitative properties of the distance .

1. Introduction

Modeling datasets as metric spaces seems to be natural for some applications and concepts revolving around the Gromov-Hausdorff distance —a notion of distance between compact metric spaces— provide a useful language for expressing properties of data and shape analysis methods.

These notes are based on a talk given during the conference "Discrete Curvature" held in Luminy in November 2013.

Notation and background concepts. The book by Burago, Burago, and Ivanov [2] is a valuable source for many concepts in metric geometry. We refer the reader to that book for any concepts not explicitly defined in these notes.

We let $\mathcal{M}$ denote the collection of all compact metric spaces. Recall that for a given metric space $(X, d_X) \in \mathcal{M}$, its *diameter* is defined as $\mathbf{diam}(X) := \max_{x,x' \in X} d_X(x, x')$. Similarly, the *radius* of X is defined as $\mathbf{rad}(X) := \min_{x \in X} \max_{x' \in X} d_X(x, x')$.

For a fixed metric space (Z, d_Z), we let $d_{\mathcal{H}}^Z$ denote the Hausdorff distance between (closed) subsets of Z.

We will often refer to a metric space (X, d_X) by only X, but the notation for the underlying metric will be implicitly understood to be d_X. Recall, that a map $\varphi : X \to Y$ between metric spaces (X, d_X) and (Y, d_Y) is an *isometric embedding* if $d_Y(\varphi(x), \varphi(x')) = d_X(x, x')$ for all $x, x' \in X$. The map φ is an *isometry* if it is a surjective isometric embedding.

2. The definition

The goal is to measure distance between two given abstract compact metric spaces. In general, these two spaces may not be readily given as subsets of a common metric space. In this case, the following construction by Gromov [4] applies.

Given (X, d_X) and (Y, d_Y) in $\mathcal{M}$ one considers any "sufficiently rich" third metric space (Z, d_Z) inside which one can find isometric copies of X and Y and measures the Hausdorff distance in Z between these copies. Finally, one minimizes over the choice of the isometric copies and Z. Formally, let Z, $\phi_X : X \to Z$ and $\phi_Y : Y \to Z$ be respectively a metric space and isometric embeddings of X and Y into Z. Then, the Gromov-Hausdorff distance between X and Y is defined as

$$(2.1) \qquad d_{\mathcal{GH}}(X, Y) := \inf_{Z, \phi_X, \phi_Y} d_{\mathcal{H}}^Z \big(\phi_X(X), \phi_Y(Y) \big).$$
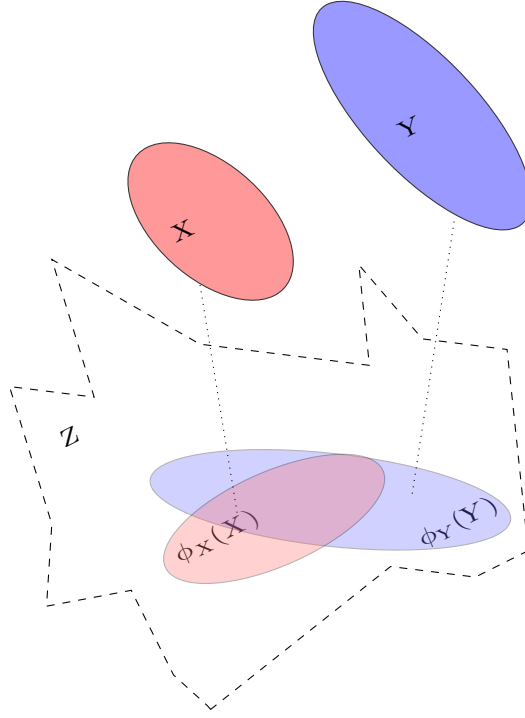
Theorem 1 ([4]). *$d_{\mathcal{GH}}$ is a legitimate distance on the collection of isometry classes of $\mathcal{M}$.*

From the practical point of view this definition might not look appealing. As we recall below, there are other more computational suggestive equivalent definitions whose implementation has been explored. But now we try to interpret the definition we have given so far.

2.1. **An example.** Consider the metric spaces X consisting exactly of three points at distance 1 from each other, and Y consisting of exactly one point. Notice that X and Y can be *simultaneously* embedded into $\mathbb{R}^2$ in an isometric way so that $Z = \mathbb{R}^2$ is a valid choice in (2.1) above. The maps ϕ_X and ϕ_Y represent the relative positions of X and Y in the plane.

By homogeneity, we can assume that the embedding of X is fixed. When choosing ϕ_Y one notices that the optimal relative position of $q := \phi_Y(Y)$ with respect to $\Delta := \phi_X(X)$ happens when q is the center of the (equilateral) triangle Δ. In that case, the Hausdorff distance in (2.1) is $\delta_0 := \frac{1}{\sqrt{3}}$ and we conclude that $d_{\mathcal{GH}}(X, Y) \leq \delta_0$. One would be tempted to think that δ_0 is in fact *equal* to Gromov-Hausdorff distance between X and Y but this is not the case!

The same construction that we did above for $\mathbb{R}^2$ can in fact be done on the model hyperbolic two-dimensional space $\mathbb{H}_\kappa$ of curvature $-\kappa$ for *any* $\kappa \leq 0$. As $\kappa \to -\infty$, the (geodesic interpolation of the) triangle Δ becomes 'thinner' and intuitively, the Hausdorff distance δ_κ between the optimal embeddings in $\mathbb{H}_\kappa$ will decrease as κ decreases.

One can in fact consider the following target metric space: Z_∞ consists of four points p_1, p_2, p_3, and p such that $d_Z(p_i, p_j) = 1$ for $i \neq j$ and $d_Z(p_i, p) = \frac{1}{2}$ for all i. This metric space with four points can be regarded as a subset of the *real tree* (geodesic) metric space below:
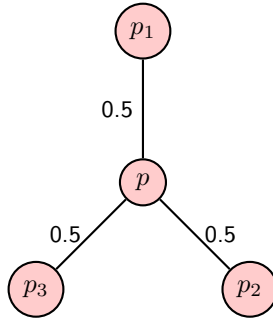
This metric space can be regarded as an extreme case of the construction involving the $\mathbb{H}_\kappa$ that was described above. The interesting fact is that if we let $\phi_X(X) = \{p_1, p_2, p_3\}$ and $\phi_Y(Y) = \{q\}$, then $\delta_\infty := d_\mathcal{H}^{Z_\infty}\big(\phi_X(X), \phi_Y(Y)\big) = \frac{1}{2}$ which is strictly smaller than δ_0! and thus proves that

$$d_{\mathcal{GH}}(X, Y) \leq \frac{1}{2} < \frac{1}{\sqrt{3}}.$$

One can in fact check that $\delta_\infty < \delta_\kappa \leq \delta_0$ for all $\kappa \in [0, \infty)$. In any case, as we recall in Corollary 5 below, $d_{\mathcal{GH}}(X, Y)$ is always bounded below by $\frac{1}{2}|\mathbf{diam}(X) - \mathbf{diam}(Y)|$. Since in the present case $\mathbf{diam}(X) = 1$ and $\mathbf{diam}(Y) = 0$, we obtain that $d_{\mathcal{GH}}(X, Y) \geq \frac{1}{2}$ which together with the reverse inequality obtained above implies that in fact, for the example under consideration, $d_{\mathcal{GH}}(X, Y) = \frac{1}{2}$!

2.2. **A simplification.** Kalton and Ostrovskii [5] observed that one can *equivalently* define the Gromov-Hausdorff distance between X and Y by considering Z in (2.1) to be the disjoint union $X \sqcup Y$ together with *any* metric d such that $\mathrm{d}_{|X \times X} = d_X$ and $\mathrm{d}_{|Y \times Y} = d_Y$. Let $\mathcal{D}(d_X, d_Y)$ denote the set of *all* such metrics on $X \sqcup Y$. Then, they observe that

$$(2.2) \qquad d_{\mathcal{GH}}(X, Y) = \inf_{\mathrm{d} \in \mathcal{D}(d_X, d_Y)} d_\mathcal{H}^{(X \sqcup Y, \mathrm{d})}(X, Y).$$

This expression for the Gromov-Hausdorff distance seems more appealing for the computationally minded: imagine that X and Y are finite, then the *variable* d in the underlying optimization problem can be regarded as a matrix in $\mathbb{R}^{|X| \times |Y|}$. If we assume that $|X| = |Y| = n$ then the number of *linear constraints* that each d in $\mathcal{D}(d_X, d_Y)$ must satisfy is of order n^3 (all triangle inequalities). Even more explicitly, the optimization problem over $\mathcal{D}(d_X, d_Y)$ that one must solve in practice is (cf. [7]) $\min_{\mathrm{d}} J(\mathrm{d})$ where

$$J(\mathrm{d}) := \max\left(\max_{x \in X} \min_{y \in Y} \mathrm{d}(x, y), \max_{y \in Y} \min_{x \in X} \mathrm{d}(x, y) \right).$$

The complexity from the original definition (2.1) is now hidden in the fact that $J(\cdot)$ is highly non-linear.

Going back to the example discussed in 2.1, one can state that in the context of (2.2), the optimal metric on $X \sqcup Y$ is

$$\mathrm{d}^* := \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 1 \\ \frac{1}{2} & 0 & \frac{1}{2} & 1 \\ \frac{1}{2} & \frac{1}{2} & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

2.3. **The case of subsets of Euclidean space.** Even if we saw in Section 2.1 above that when X and Y are subsets of $\mathbb{R}^d$ the optimal Z in (2.1) may not be $\mathbb{R}^d$, one can still relate $d_{\mathcal{GH}}(X, Y)$ with some natural notion of distance for subsets of Euclidean space. Doing this provides more insight into as to how the Gromov-Hausdorff distance operates in situations for which we already have a well developed intuition.

An *intrinsic approach* to comparing two subsets X and Y of $\mathbb{R}^d$ would be to regard them as metric spaces by endowing them with the restriction of the ambient space metric: $d_X(\cdot, \cdot) = \|\cdot - \cdot\|$ etc. So, one can consider $d_{\mathcal{GH}}(X, Y)$ as a possible notion of dissimilarity between X and Y.

Another notion of dissimilarity that is frequently considered in shape and data analysis arises from the Hausdorff distance modulo rigid isometries and constitutes an *extrinsic approach*: let $E(d)$ denote the group of isometries of $\mathbb{R}^d$ and define

$$d_\mathcal{H}^{\mathbb{R}^d, \text{rigid}}(X, Y) := \inf_{T \in E(d)} d_\mathcal{H}^{\mathbb{R}^d}\big(X, T(Y)\big).$$

Since in this case, one can always choose $Z = \mathbb{R}^d$ in (2.1) above, one immediately sees that $d_{\mathcal{GH}}(X, Y) \leq d_\mathcal{H}^{\mathbb{R}^d, \text{rigid}}(X, Y)$. Even if we already saw in Section 2.1 that the equality cannot take place in general, one could hope that for some suitable $C > 0$, $d_\mathcal{H}^{\mathbb{R}^d, \text{rigid}}(X, Y) \leq C \cdot d_{\mathcal{GH}}(X, Y)$ for all $X, Y \subset \mathbb{R}^d$ compact. Interestingly, however, this cannot happen! Consider $X = [-1, 1]$. Fix $0 < \varepsilon \ll 1$ and let $f_\varepsilon(x) := |x| \cdot \sqrt{\varepsilon}$. Let Y_ε be the set $\{(x, f_\varepsilon(x)); x \in [-1, 1]\}$. Notice that $\mathbf{rad}(X) = 1$ and $\mathbf{rad}(Y_\varepsilon) = \sqrt{1 + \varepsilon}$.
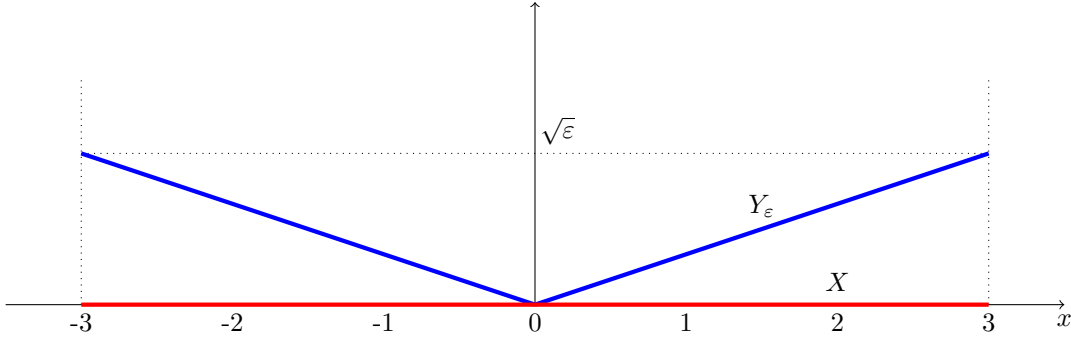
In any case, it is clear that for $\varepsilon > 0$ small enough, $d_{\mathcal{H}}^{\mathbb{R}^d,\mathrm{rigid}}(X, Y_\varepsilon) = \frac{\sqrt{\varepsilon}}{2}$. However, since by Proposition 6 and Corollary 4 below,

- $d_{\mathcal{GH}}(X, Y_\varepsilon) \geq \frac{1}{2}|\mathbf{rad}\,(X) - \mathbf{rad}\,(Y_\varepsilon)| = \frac{1}{2}(\sqrt{1+\varepsilon} - 1) \geq \frac{\varepsilon}{2+2\sqrt{2}}$ and

- $d_{\mathcal{GH}}(X, Y_\varepsilon) \leq \frac{1}{2}\sup_{|x|\neq|x'|}|x-x'|\cdot\left(\sqrt{1 + \varepsilon \cdot \left(\frac{|x|-|x'|}{x-x'}\right)^2} - 1\right) \leq \varepsilon$, since $\big||x|-|x'|\big| \leq |x-x'|$ for all $x, x' \in X$.

It follows that $d_{\mathcal{GH}}(X, Y_\varepsilon)$ is of order ε and therefore no constant $C > 0$ will guarantee that $C \cdot d_{\mathcal{GH}}(X, Y_\varepsilon) \geq d_{\mathcal{H}}^{\mathbb{R}^d,\mathrm{rigid}}(X, Y_\varepsilon)$ for all $1 \gg \varepsilon > 0$!

What does hold for this construction is that $C \cdot \big(d_{\mathcal{GH}}(X, Y_\varepsilon)\big)^{1/2} \geq d_{\mathcal{H}}^{\mathbb{R}^d,\mathrm{rigid}}(X, Y_\varepsilon)$ for some constant $C > 0$. It turns out that this is not an isolated phenomenon:

Theorem 2 ([6])**.** *For each natural number $d \geq 2$ there exists $c_d > 0$ such that for all $X, Y \in \mathbb{R}^d$ one has*
$$d_{\mathcal{GH}}(X, Y) \leq d_{\mathcal{H}}^{\mathbb{R}^d,\mathrm{rigid}}(X, Y) \leq c_d \cdot M^{1/2} \cdot \big(d_{\mathcal{GH}}(X, Y)\big)^{1/2},$$
where $M = \max(\mathbf{diam}\,(X), \mathbf{diam}\,(Y))$.

2.4. **Another expression and consequences.** For two sets X and Y let $\mathcal{R}(X, Y)$ denote the set of all *correspondences* between X and Y, that is, sets $R \subseteq X \times Y$ such that $\pi_1(R) = X$ and $\pi_2(R) = Y$. In general, we will refer to any non-empty set R of $X \times Y$ as a *relation* between X and Y. Obviously, all correspondences are relations.

The *distortion* of a relation R between the metric spaces (X, d_X) and (Y, d_Y) is defined as the number
$$\mathrm{dis}(R) := \sup_{(x,y),(x',y')\in R} \big|d_X(x, x') - d_Y(y, y')\big|.$$

Notice that given a function $\varphi : X \to Y$ one can define the relation $R_\varphi := \{(x, \varphi(x)); x \in X\}$, and in that case we write $\mathrm{dis}(\varphi) := \mathrm{dis}(R_\varphi) = \sup_{x,x'\in X}\big|d_X(x, x') - d_Y(\varphi(x), \varphi(x'))\big|$. Similarly, when $\psi : Y \to X$ is given, it induces the relation $R_\psi := \{(\psi(y), y); y \in Y\}$. Note that the structure of R_φ is different from the structure of R_ψ.

Now, when a map $\varphi : X \to Y$ and a map $\psi : Y \to X$ are both specified, we consider the relation $R_{\varphi,\psi} := R_\varphi \bigcup R_\psi$ and note that in fact $R_{\varphi,\psi}$ is actually a correspondence between X and Y.

Furthermore, one can explicitly compute that
$$\mathrm{dis}(R_{\varphi,\psi}) = \max\big(\mathrm{dis}(\varphi), \mathrm{dis}(\psi), C(\varphi, \psi)\big),$$
where $C(\varphi, \psi) := \sup_{x\in X, y\in Y}\big|d_X(x, \psi(y)) - d_Y(\varphi(x), y)\big|$. Notice that if $C(\varphi, \psi) < \eta$ for some $\eta > 0$, then $\big|d_X(x, \psi(y)) - d_Y(\varphi(x), y)\big| < \eta$ for all $(x, y) \in X \times Y$. In particular, for $x = \psi(y)$, it follows that $d_Y(\varphi \circ \psi(y), y) < \eta$ for all $y \in Y$. Similarly one can obtain $d_X(x, \psi \circ \varphi(x)) < \eta$ for all $x \in X$. These two conditions are often interpreted as meaning that φ and ψ are close to being inverses of each other. This proximity is quantified by η.

An interesting and useful characterization of the Gromov-Hausdorff distance based on optimization over correspondences is the following:

Theorem 3 ([5])**.** *For all $X, Y \in \mathcal{M}$ one has that*
$$d_{\mathcal{GH}}(X, Y) \stackrel{(\mathrm{I})}{=} \frac{1}{2}\inf_{R\in\mathcal{R}(X,Y)} \mathrm{dis}(R) \stackrel{(\mathrm{II})}{=} \frac{1}{2}\inf_{\varphi,\psi} \mathrm{dis}(R_{\varphi,\psi}).$$

Corollary 4. *Let X be a set and d and d' be any two metrics on X. Then,*

$$d_{\mathcal{GH}}((X,d),(X,d')) \leq \frac{1}{2} \sup_{x,x' \in X} |d(x,x') - d'(x,x')|.$$

The theorem above is significant for several reasons. First of all, (I) indicates that solving for the Gromov-Hausdorff distance between two finite metric spaces is an instance of a well known combinatorial optimization problem called the *bottleneck quadratic assignment problem* or bQAP. The bQAP is NP-Hard and furthermore, computing any $(1 + \varepsilon)$ of the optimal solution is also NP-Hard for any $\varepsilon > 0$ [12]. See [9, 10, 1] for some heuristic approaches.

A second observation stemming from the equality (II) in the theorem is the fact that since the term $C(\varphi,\psi)$ acts as a *coupling term* in the optimization

$$d_{\mathcal{GH}}(X,Y) = \frac{1}{2} \inf_{\varphi,\psi} \max\big(\mathrm{dis}(\varphi), \mathrm{dis}(\psi), C(\varphi,\psi)\big),$$

one could conceive of dropping it from the expression above yielding

$$d_{\mathcal{GH}}(X,Y) \geq \frac{1}{2} \max\big(\inf_{\varphi} \mathrm{dis}(\varphi), \inf_{\psi} \mathrm{dis}(\psi)\big) =: \widehat{d}_{\mathcal{GH}}(X,Y).$$

It is important to notice that computing $\widehat{d}_{\mathcal{GH}}(X,Y)$, which we call the *modified Gromov-Hausdorff distance* [8], leads to solving two *decoupled* optimization problems, a feature which is desirable in applications. However, the computational complexity of the problems of the type $\inf_\varphi \mathrm{dis}(\varphi)$ could still be high. We will explore some interesting structure that arises from this modified definition in the next section but for now we will make one more observation based on the expression given by Theorem 3.

From equality (I) it follows that the Gromov-Hausdorff distance between any compact metric space and the metric space consisting of exactly one point is $d_{\mathcal{GH}}(X, *) = \frac{1}{2}\mathbf{diam}(X)$. As a corollary from Theorem 1 and this observation one has

Corollary 5. *For all $X, Y \in \mathcal{M}$, $d_{\mathcal{GH}}(X,Y) \geq \frac{1}{2}\big|\mathbf{diam}(X) - \mathbf{diam}(Y)\big|$.*

Proof. The inequality $d_{\mathcal{GH}}(X,Y) \geq \big|d_{\mathcal{GH}}(X,*) - d_{\mathcal{GH}}(Y,*)\big|$ is guaranteed by the triangle inequality for the Gromov-Hausdorff distance. The remark preceding the statement completes the proof. $\square$

A similar lower bound for the Gromov-Hausdorff distance arises from considering the radius of metric spaces:

Proposition 6 ([8])**.** *For all $X, Y \in \mathcal{M}$, $d_{\mathcal{GH}}(X,Y) \geq \frac{1}{2}\big|\mathbf{rad}(X) - \mathbf{rad}(Y)\big|$.*

3. The modified Gromov-Hausdorff and curvature sets

It could appear plausible that by dropping the coupling term $C(\varphi,\psi)$ in the optimization above one might have lost some of the nice theoretical properties enjoyed by the Gromov-Hausdorff distance. This is not the case, and in fact the modified Gromov-Hausdorff retains many of these good properties:

Theorem 7 ([8])**.** *The modified Gromov-Hausdorff distance satisfies:*

    (1) *$\widehat{d}_{\mathcal{GH}}$ is a legitimate metric on the isometry classes of $\mathcal{M}$.*
    (2) *$d_{\mathcal{GH}}(X,Y) \geq \widehat{d}_{\mathcal{GH}}(X,Y)$ for all $X, Y \in \mathcal{M}$.*
    (3) *$d_{\mathcal{GH}}$ and $\widehat{d}_{\mathcal{GH}}$ are topologically equivalent within $d_{\mathcal{GH}}$-precompact families of $\mathcal{M}$.*

It is however interesting that the equality in item (2) *does not take place in general.* In fact, [8] provides a counterexample.

3.1. **Curvature sets.** Gromov [4] defines for each $n \in \mathbb{N}$ the curvature sets of $X \in \mathcal{M}$ in the following way: let $\Psi_X^{(n)} : X^{\times n} \to \mathbb{R}^{n \times n}$ be the matrix valued map defined by $(x_1, \ldots, x_n) \mapsto ((d_X(x_i, x_j))_{i,j=1}^n$. This map simply assigns to each n-tuple of points its *distance matrix*: the matrix arising from restricting the metric on X to the given n-tuple. Then, the n-th curvature set of X is

$$\mathbf{K}_n(X) := \big\{\Psi_X^{(n)}(x_1, \ldots, x_n); (x_1, \ldots, x_n) \in X^{\times n}\big\}.$$

In colloquial terms, curvature sets are just 'bags' containing all the possible distance matrices of a given size arising from points sampled from X.

For example, when $n = 2$, $\mathbf{K}_2(X)$ contains the same information as $\{d_X(x, x'); x, x' \in X\} \subset \mathbb{R}_+$. In contrast, $\mathbf{K}_3(X)$ contains all 'triangles' from X and this particular case suggest one possible justification for the name 'curvature sets'. Indeed, let X be a smooth planar curve. Consider any three points x_1, x_2 and x_3 on X close to each other. Then, if $a = \|x_2 - x_1\|$, $b = \|x_1 - x_3\|$, and $c = \|x_1 - x_2\|$, the inverse of the radius R of the circle circumscribed to the triangle $\Delta x_1 x_2 x_3$ admits an explicit expression in terms of a, b and c: $R^{-1} = \frac{4 \, S(a,b,c)}{a \, b \, c}$ where $S(a,b,c)$ is the area of the triangle as given by Heron's formula.[1] The crucial observation is that R can be computed exclusively from the information contained in $\mathbf{K}_3(X)$. Now, by an argument involving a series expansion [3], as $a, b, c \to 0$ R^{-1} converges to the curvature κ of X at the point of coalescence of x_1, x_2, x_3.

Curvature sets absorb all the information that one needs in order to determine whether two compact metric spaces are isometric or not.

Theorem 8 ([4])**.** *Let $X, Y \in \mathcal{M}$. Then, X and Y are isometric if and only if $\mathbf{K}_n(X) = \mathbf{K}_n(Y)$ for all $n \in \mathbb{N}$.*

Constructions similar to curvature sets have also been considered by Peter Olver in the context of subsets of Euclidean space [11].

An example: Curvature sets of spheres. We illustrate the definition with an example from [8]. Consider first the case of the standard circle S^1 endowed with the angular distance. We will exactly characterize $\mathbf{K}_3(S^1)$. For that purpose first consider any embedding of S^1 into $\mathbb{R}^2$ and observe that for any three points on S^1 exactly one the following two conditions holds: (a) there exists a line through the center of the circle such that the three points are contained on one side of the line; (b) no such line exists.

Case (a) means that one of the three distances defined by the three points must forcibly be equal to the sum of the other two distances. Case (b) implies that the sum of the three distances is exactly 2π. Also note that, by symmetry, case (a) unrolls into three different cases depending on the identity of the distance that is equal to the sum of the other two. Each of these four situations gives a *linear* relation between the three distances! Thus, we obtain that $\mathbf{K}_3(S^1)$ is isomorphic to the tetrahedron with vertices $(0, 0, 0)$, $(0, \pi, \pi)$, $(\pi, 0, \pi)$, and $(\pi, \pi, 0)$.

The case of S^2, when endowed with the standard geodesic distance, is similar and one can prove that $\mathbf{K}_3(S^2)$ is the *convex hull* of $\mathbf{K}_3(S^1)$.

3.2. **Comparing curvature sets?** An interesting property of curvature sets is that they are *isometry invariants* of metric spaces which 'live' in fixed target spaces. More precisely, for any $X, Y \in \mathcal{M}$, $\mathbf{K}_n(X)$ and $\mathbf{K}_n(Y)$ are both subsets of $\mathbb{R}^{n \times n}$.

With the purpose of discriminating X and Y one may conceive of comparing $\mathbf{K}_n(X)$ and $\mathbf{K}_n(Y)$. Since they are both (compact) sub-sets of $\mathbb{R}^{n \times n}$ one could compute the Hausdorff distance between them. For this we first endow $\mathbb{R}^{n \times n}$ with the distance $d_{\ell^\infty}(A, B) := \max_{i,j} |a_{i,j} - b_{i,j}|$ for $A = ((a_{i,j}))$ and $B = ((b_{i,j}))$ in $\mathbb{R}^{n \times n}$. Then, we compute

$$d_n(X, Y) := \frac{1}{2} \, d_{\mathcal{H}}^{\mathbb{R}^{n \times n}}\big(\mathbf{K}_n(X), \mathbf{K}_n(Y)\big),$$

and use this number as an indication of how similar X and Y are. The best possible measure of dissimilarity that this sort of idea suggests is to consider

$$d_\infty(X, Y) := \sup_{n \in \mathbb{N}} d_n(X, Y).$$

Theorem 8 guarantees that d_∞ defines a legitimate metric on $\mathcal{M}$ modulo isometries.

Interestingly, one has the following 'structural theorem' for the modified Gromov-Hausdorff distance in terms of curvature sets:

Theorem 9 ([8])**.** *For all $X, Y \in \mathcal{M}$, $\widehat{d}_{\mathcal{GH}}(X, Y) = d_\infty(X, Y)$.*

---

[1] $S(a, b, c) = \frac{1}{4}\big((a + b + c)(a - b + c)(a + b - c)(-a + b + c)\big)^{1/2}$.

This theorem provides a useful path for computing estimates to the Gromov-Hausdorff distance. Furthermore, the theorem suggests a way of 'slicing' the computation/approximation of the Gromov-Hausdorff distance between finite metric spaces, since one might want to consider computing d_n for a fixed n and hope that this provides enough information for discriminating spaces within a given family. For finite spaces, the computation of d_n would incur a polynomial cost, albeit of a high order. There are some known classes of metric spaces $\mathcal{C} \subset \mathcal{M}$ that are characterized up to isometry by $\mathbf{K}_n(\cdot)$ for some finite $n = n(\mathcal{C})$, see [8].

A lower bound for $d_{\mathcal{G}\mathcal{H}}(S^1, S^2)$. Theorems 7 item (2) and 9 then guarantee that

$$d_{\mathcal{G}\mathcal{H}}(S^1, S^2) \geq d_3(S^1, S^2) = \frac{1}{2} d_{\mathcal{H}}^{\mathbb{R}^{3\times3}}\left(\mathbf{K}_3(S^1), \mathbf{K}_3(S^2)\right) =: \xi.$$

Since $\mathbf{K}_3(S^2)$ is the convex hull of $\mathbf{K}_3(S^1)$, $\mathbf{K}_3(S^1) \subset \mathbf{K}_3(S^2)$, and therefore,

$$\xi = \frac{1}{2} \max_{p\in\mathbf{K}_3(S^2)} \min_{q\in\mathbf{K}_3(S^1)} \|p - q\|_\infty = \min_{q\in\mathbf{K}_3(S^1)} \|g - q\|,$$

where $g = \frac{\pi}{2}(1,1,1)$ is the center of $\mathbf{K}_3(S^2)$. But now, the center $c = \frac{2\pi}{3}(1,1,1)$ of the face of $\mathbf{K}_3(S^1)$ determined by $\pi(0,1,1)$, $\pi(1,0,1)$, and $\pi(1,1,0)$ is at minimal ℓ^∞ distance from g so that $\xi = \frac{1}{2}|\frac{\pi}{2} - \frac{2\pi}{3}| = \frac{\pi}{12}$, and we find the lower bound $d_{\mathcal{G}\mathcal{H}}(S^1, S^2) \geq \frac{\pi}{12}$.

4. Discussion and outlook

The Gromov-Hausdorff distance offers a useful language for expressing different tasks in shape and data analysis. Its origins are in the work of Gromov on synthetic geometry. For finite metric spaces, the Gromov-Hausdorff distance leads to solving NP-Hard combinatorial optimization problems. A related to construction is that of *Gromov-Wasserstein* distances which operate on metric measure spaces [13, 7]. In contrast to the Gromov-Hausdorff distance, the computation of Gromov-Wasserstein distances leads to solving quadratic optimization problems on *continuous* variables. The space of all metric measures spaces endowed with a certain variant of the Gromov-Wasserstein distance [7] enjoys nice theoretical properties [14]. It seems of interest to develop provably correct approximations to these distances when restricted to some suitable subclasses of finite metric spaces.

References

[1] Alexander M. Bronstein, Michael M. Bronstein, and Ron Kimmel. Efficient computation of isometry-invariant distances between surfaces. *SIAM Journal on Scientific Computing*, 28(5):1812–1836, 2006.

[2] D. Burago, Y. Burago, and S. Ivanov. *A Course in Metric Geometry*, volume 33 of *AMS Graduate Studies in Math.* American Mathematical Society, 2001.

[3] Eugenio Calabi, Peter J. Olver, Chehrzad Shakiban, Allen Tannenbaum, and Steven Haker. Differential and numerically invariant signature curves applied to object recognition. *Int. J. Comput. Vision*, 26(2):107–135, 1998.

[4] Misha Gromov. *Metric structures for Riemannian and non-Riemannian spaces*, volume 152 of *Progress in Mathematics*. Birkhäuser Boston Inc., Boston, MA, 1999.

[5] N. J. Kalton and M. I. Ostrovskii. Distances between Banach spaces. *Forum Math.*, 11:1:17–48, 1999.

[6] Facundo Mémoli. Gromov-Hausdorff distances in Euclidean spaces. *Computer Vision and Pattern Recognition Workshops, 2008. CVPR Workshops 2008. IEEE Computer Society Conference on*, pages 1–8, June 2008.

[7] Facundo Mémoli. Gromov-Wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11:417–487, 2011.

[8] Facundo Mémoli. Some properties of Gromov—Hausdorff distances. *Discrete & Computational Geometry*, pages 1–25, 2012. 10.1007/s00454-012-9406-8.

[9] Facundo Mémoli and Guillermo Sapiro. Comparing point clouds. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 32–40, New York, NY, USA, 2004. ACM.

[10] Facundo Mémoli and Guillermo Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Found. Comput. Math.*, 5(3):313–347, 2005.

[11] P.J. Olver. Joint invariant signatures. *Foundations of computational mathematics*, 1(1):3–68, 2001.

[12] Panos M. Pardalos and Henry Wolkowicz, editors. *Quadratic assignment and related problems*. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 16. American Mathematical Society, Providence, RI, 1994. Papers from the workshop held at Rutgers University, New Brunswick, New Jersey, May 20–21, 1993.

[13] Karl-Theodor Sturm. On the geometry of metric measure spaces. I. *Acta Math.*, 196(1):65–131, 2006.

[14] Karl-Theodor Sturm. The space of spaces: curvature bounds and gradient flows on the space of metric measure spaces. *arXiv preprint arXiv:1208.0434*, 2012.

Department of Mathematics, The Ohio State University, Columbus, OH 43210, United States of America • memoli@math.osu.edu

Curvature on a graph via its geometric spectrum

Paul Baird

Abstract

We approach the problem of defining curvature on a graph by attempting to attach a 'best-fit polytope' to each vertex, or more precisely what we refer to as a configured star. How this should be done depends upon the global structure of the graph which is reflected in its geometric spectrum. Mean curvature is the most natural curvature that arises in this context and corresponds to local liftings of the graph into a suitable Euclidean space. We discuss some examples.

1. Introduction

The problem we address is one of ascribing geometry to a graph using just its combinatorial structure. Geometry should *emerge* from the structure rather than being imposed upon it. Our approach is to appeal to the way we perceive objects in the world around us. A good starting point is what psychologists refer to as the Necker cube.
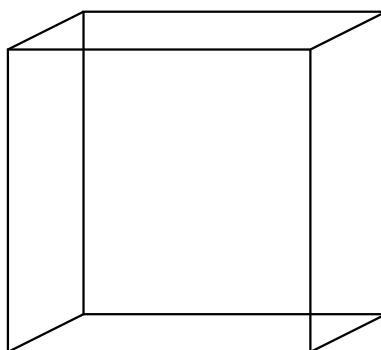
When we see this picture we generally perceive one of two possible 3-dimensional cubes. The reasons for this are a matter of cognitive science, but also the way the graph has been drawn on the piece of paper. With the graph so realized as a *framework* in 3-dimensional Euclidean space, we can begin to do geometry: edge length is defined; the Gauss curvature is defined at each vertex in terms of the angular deficit; other curvatures appear, such as the rotation of the Gauss map between adjacent vertices. But how has the realization of this graph come about?

To perform this procedure on an more general graph, we attempt to produce a "best-fit polytope" at each vertex, or rather, what we call a *configured star*, by lifting a vertex and its neighbours in a natural way into a Euclidean space $\mathbb{R}^N$. A configured star generalizes the star framework at each vertex of a regular polytope. More specifically, it consists of an internal vertex connected to n external vertices that have a particularly symmetric configuration (see below). In particular, any

configured star has an axis which can be interpreted as the Gauss map at a vertex. The dimension N of the space into which we lift is a matter of choice but must always be less than or equal to the degree n of the vertex.

But now if we return to the cube pictured above, how do we decide whether to fit a cube to each vertex, or a regular tetrahedron, since both have vertices of degree 3 ? The answer is the cube because of the *global* combinatorial structure of the graph. In order to bring this global structure into play, we introduce the notion of *geometric spectrum* in [3, 1]. The geometric spectrum is a real-valued parameter γ defined on the vertices of a graph for which the quadratic difference equation :

$$(1.1) \qquad\qquad \gamma(\Delta\phi)^2 = (\nabla\phi)^2$$

has a non-trivial solution ϕ (see (2.2) below for the explicit form of this equation). For technical reasons that we explain below, we require that $\gamma < 1$. The smooth version of equation (1.1) applied to a hypersurface in Euclidean space shows that $1/\gamma = -H^2$ where H is the mean curvature. Since for a mesh which approximates a smooth hypersurface, equation (1.1) approximates its smooth counterpart, it is reasonable to consider γ as corresponding to mean curvature (or more precisely to $-1/H^2$). It is worth noting that for regular convex polyhedra, equation (1.1) is satisfied with γ positive in the case of a small number of vertices, for example the tetrahedron, and then becoming negative when the number of vertices increases, for example for the icosahedron, the dodecahedron and the 4-dimensional 600-cell, as the polyhedron approximates better a smooth round sphere.

For a complete graph on $N + 1$ vertices, the geometric spectrum consists of a single value $N/(N + 1)$ and the solution ϕ corresponds to the images of the vertices after an orthogonal projection of the regular N-simplex to the complex plane. At the other extreme, for a cyclic graph on n vertices, solutions ϕ to (1.1) correspond to realizations of the graph as an n-polygon in the plane with sides of equal length. Now the geometric spectrum has continuous components with complicated branching phenomena. An interpretation of the geometric spectrum as *information* implicit in a graph which may be exploited to enact structural change is discussed in the article [4].

Given a solution to (1.1) on a graph with $\gamma < 1$, it is shown in [5] how one has a local lifting property at each vertex: each vertex and its neighbours can be lifted to an invariant configured star in Euclidean space $\mathbb{R}^N$ with $2 \leq N \leq n$, where n is the degree of the vertex. In the case when $N = 3$, except for some special cases, this lifting is unique up to a one of two possibilities (for example, the two visualizations of the cube), as such, this dimension becomes the most interesting when degrees are ≥ 3. How we choose from the two possible liftings is then a global matter of correlating liftings at adjacent vertices (cf. the cube). For example, an Escher picture has local liftings with a global inconsistency.

Given a graph and a solution to (1.1), we can now make sense of distance and curvature. In general, provided we are only interested in relative distance between one part of the graph and another and curvature which doesn't depend on scale (for example Ricci curvature), these should depend only on the geometric spectrum γ and not on the solution ϕ.

2. The geometric spectrum

Any regular polytope in Euclidean space satisfies the following quadratic difference equation:

$$(2.1) \qquad\qquad \frac{\gamma}{n}\Big(\sum_{y\sim x} \big(\phi(y) - \phi(x)\big) \Big)^2 = \sum_{y\sim x} \big(\phi(y) - \phi(x)\big)^2,$$

at each of its vertices x, where ϕ is an orthogonal projection to the complex plane, n is the (common) degree at each vertex (the number of edges incident with x) and $y \sim x$ means that y is connected to x by an edge. This fact is implicit in the work of Eastwood and Penrose [10] and made explicit by the author in [5]. The constant γ depends on the polytope and it can be either positive or negative but it is always < 1. For the convex regular polyhedra in $\mathbb{R}^3$ (the Platonic solids), the values of γ can be calculated by expressing the angular deficit at each vertex in terms of γ (cf. [1], Proposition 7.4) and then applying the classical theorem of Descartes: *for a convex polyhedron in $\mathbb{R}^3$, the sum over the vertices of their angular deficits is equal to 4π.* See table 2.1. Note that the value of γ in (2.1) is invariant by any similarity transformation of the polytope, which suggests that the intrinsic geometry of the polytope may be characterized by three aspects:

polyhedron	γ
tetrahedron	$3/4$
cube	0
octahedron	$1/2$
icosahedron	$\frac{2-\sqrt{5}}{3-\sqrt{5}} < 0$
dodecahedron	$\frac{3(1-\sqrt{5})}{2(3-\sqrt{5})} < 0$

Table 2.1: Values of γ for the Platonic solids.

(i) the fact that the underlying framework (or 1-skeleton) satisfies (2.1) ; (ii) the value of the constant γ ; (iii) the underlying combinatorial structure. Let us therefore proceed to generalize this construct to a more general graph.

Given a graph $\Gamma = (V, E)$ with vertex set V and edge set E, together with a real-valued function $\gamma : V \to \mathbb{R}$, we introduce the equation:

$$(2.2) \qquad \frac{\gamma(x)}{n(x)} \Big(\sum_{y \sim x} \big(\phi(y) - \phi(x) \big) \Big)^2 = \sum_{y \sim x} \big(\phi(x) - \phi(y) \big)^2 ,$$

at each vertex x, where $\phi : V \to \mathbb{C}$ is a complex-valued function and $n(x)$ is the degree of Γ at x. Solutions with $\gamma \equiv 0$ have been called *holomorphic functions*[1] and have been used to give a description of massless fields in a combinatorial setting [6]. Note that the equations are invariant by the replacement of ϕ by the transformations

$$(2.3) \qquad \phi \mapsto \lambda \phi + \mu \quad (\lambda, \mu \in \mathbb{C}), \quad \text{and} \quad \phi \mapsto \overline{\phi}.$$

We shall consider two solutions related in this way as *equivalent*.

If we set $\Delta \phi(x) = \frac{1}{n(x)} \sum_{y \sim x} \big(\phi(y) - \phi(x) \big)$ (the Laplacian) and $(\nabla \phi)^2(x) = \frac{1}{n(x)} \sum_{y \sim x} \big(\phi(y) - \phi(x) \big)^2$ (the symmetric square derivative), then equation (2.2) has the more economic expression (1.1) given in the Introduction. The Cauchy-Schwarz inequality shows that for a given vertex x if the values $\{ \phi(y) - \phi(x) : y \sim x \}$ are real and not all zero, then $\gamma(x) \geq 1$ [5].

For a given graph, we would like to know what are the admissible functions $\gamma : V \to \mathbb{R}$ for which (2.2) has a solution. Define the *geometric spectrum* of Γ to be the collection of equivalence classes of functions:

$$\Sigma = \{ \gamma : V \to [-\infty, 1) \subset \mathbb{R} : \exists \text{ non} - \text{const. } \phi : V \to \mathbb{C} \text{ satisfying (2.2)} \} ,$$

where two functions are identified when they determine a common solution ϕ and agree on the compliment of the set $\{ x \in V : \Delta \phi(x) = (\nabla \phi)^2(x) = 0 \}$. The upper bound on γ is a consequence of the Cauchy-Schwarz inequality and our requirement of invariance with respect to similarity transformations. We allow γ to take on the value $-\infty$ at points where the Laplacian vanishes.

By a *framework* in Euclidean space, we mean a graph that is realized as a subset of Euclidean space with edges straight line segments joining the vertices. We say that it is *immersed* if all vertices are distinct and *embedded* if it is immersed and edges only intersect at end points. The framework is called *invariant* if for a particular γ, it satisfies (2.2) with ϕ the restriction to the vertices of some orthogonal projection to the complex plane *independently of any similarity transformation of the framework*.

Questions that now arise are:

• For a given graph Γ, what is its geometric spectrum?

• Does a solution to (2.2) arise from an embedding of the graph as an invariant framework in Euclidean space?

• Even if the answer to the last question is no, can we still define geometric quantities such as edge length and curvature from a solution?

• To what extent do such quantities depend only on γ rather than on the choice of solution ϕ?

---

[1]A notion of *holomorphic function* somewhat similar to this has been introduced by S. Barré [7]; however, in addition to (2.2) with $\gamma \equiv 0$, Barré requires that ϕ be harmonic.

For an arbitrary graph, the geometric spectrum is determined by a fairly complicated set of algebraic equations. For graphs of sufficiently small order, these can be solved by the method of Gröbner bases with MAPLE, see [1]. The more connected a graph, the more restricted is its geometric spectrum as discussed in the Introduction.

In order to interpret γ, we consider equation (1.1) in the smooth case of a hypersurface in Euclidean space and find an interesting connection with mean-curvature.

Theorem 1. ([2]) *Let M^n be a smooth hypersurface in $\mathbb{R}^{n+1}$ ($n \geq 1$) and let g denote the metric on M^n induced from the standard metric on $\mathbb{R}^n$. Let $\phi : (M^n, g) \to \mathbb{C}$ be any orthogonal projection; then*

$$(2.4) \qquad (\Delta\phi)^2 = -H^2(\nabla\phi)^2,$$

where H is the mean curvature of M^n, and where in local coordinates, $\Delta\phi = g^{ij}(\phi_{ij} - \Gamma_{ij}^k \phi_k)$ and $(\nabla\phi)^2 = g^{ij}\phi_i\phi_j$ (summing over repeated indices).

In the case when $n = 1$, the theorem confirms the identity

$$c''(s) = \kappa(s)ic'(s),$$

for a regular curve $c : I \subset \mathbb{R} \to \mathbb{C}$ parametrized with respect to arc length. It is necessary that M^n be a *hypersurface* in order to satisfy (1.1). For example, consider the surface in $\mathbb{R}^4$ parametrized in the form:

$$(x^1, x^2) \mapsto (x^1, x^2, x^1 x^2, x^1 + x^2).$$

Let $\phi : \mathbb{R}^4 \to \mathbb{C}$ be the projection $\phi(x^1, x^2, x^3, x^4) = x^1 + x^2 i$. Then is is readily checked that the function γ defined by (1.1) is not even real.

Given the above theorem, we expect an invariant framework that closely coincides with a smooth hypersurface to have γ approximately equal to $-1/H^2$ modulo a scaling factor (equation (2.4) is not scale invariant; in order to make it so, a volume term should be added).

3. Configured stars and the lifting problem

A star graph, or bipartite graph $K_{1,n}$, has one internal vertex connected to n external vertices; there are no other connections. A star framework in $\mathbb{R}^N$ with internal vertex located at the origin can be specified by a $(N \times n)$–matrix W whose columns are the components of the external vertices. We will refer to W as the *star matrix*. Provided the centre of mass of the external vertices does not coincide with the origin, then it defines a line through the origin which we refer to as the *axis of the star*. We are interested in a particular class of star frameworks whose external vertices form what we call a configuration in a plane orthogonal to the axis of the star.

A collection of points $\{\vec{v}_1, \ldots, \vec{v}_n\}$ in $\mathbb{R}^{N-1}$ forms a *configuration* if the $((N-1) \times n)$-matrix $U = (\vec{v}_1|\vec{v}_2|\cdots|\vec{v}_n)$ whose columns have as components the coordinates $v_{\ell j}$ of $\vec{v}_\ell$ ($j = 1, \ldots, N-1$; $\ell = 1, \ldots, n$), satisfies:

$$(3.1) \qquad UU^t = \rho I_{N-1}, \qquad \sum_{\ell=1}^{n} \vec{v}_\ell = \vec{0},$$

for some non-zero constant ρ (necessarily positive), where $\vec{0}$ denotes the zero vector in $\mathbb{R}^{N-1}$ and U^t denotes the transpose of U. Necessarily, $\text{rank}(U) = N - 1$ so that $n \geq N$. A star in $\mathbb{R}^N$ whose external vertices form a configuration in a plane not passing through the origin, is referred to as a *configured star*. An *invariant* of such a star is a quantity that is invariant by orthogonal transformation. The following lemma characterizes configured stars [5].

Lemma 2. *Consider a configured star in $\mathbb{R}^N$ ($N \geq 2$) with internal vertex the origin connected to n external vertices $\{\vec{x}_1, \ldots, \vec{x}_n\}$ ($n \geq N$). Let $W = (\vec{x}_1|\vec{x}_2|\cdots|\vec{x}_n)$ be the $(N \times n)$-matrix whose columns are the components $x_{\ell j}$ of $\vec{x}_\ell$ ($j = 1, \ldots, N$; $\ell = 1, \ldots, n$). Then*

$$(3.2) \qquad WW^t = \rho I_N + \sigma \vec{u}\vec{u}^t, \qquad \sum_{\ell=1}^{n} \vec{x}_\ell = \sqrt{n(\sigma + \rho)}\, \vec{u},$$

where $\vec{u} \in \mathbb{R}^N$ is a unit vector called the axis of the star, *$\rho > 0$ and $\rho + \sigma > 0$. The quantities n, ρ, σ are all invariants of the star; the vector $\vec{u}$ is normal to the affine plane containing $\vec{x}_1, \ldots, \vec{x}_n$.*

Conversely, any matrix $W = (\vec{x}_1|\vec{x}_2|\cdots|\vec{x}_n)$ satisfying (3.2) determines a configured star with internal vertex the origin and external vertices $\vec{x}_1, \ldots, \vec{x}_n$.

Corollary 3. *Let $W = (\vec{x}_1|\vec{x}_2|\cdots|\vec{x}_n)$ define a configured star and let $\phi : \mathbb{R}^N \to \mathbb{C}$ be orthogonal projection $\phi(y_1, y_2, \ldots, y_N) = y_1 + iy_2$. Then if $z_\ell = \phi(\vec{x}_\ell) = x_{\ell 1} + ix_{\ell 2}$, we have*

$$\frac{\sigma}{n(\sigma + \rho)} \left(\sum_{\ell=1}^n z_\ell \right)^2 = \sum_{\ell=1}^n z_\ell^2 \,,$$

where ρ and σ are given by (3.2). In particular, with reference to equation (2.2), $\gamma = \sigma/(\sigma + \rho)$ is real and depends only on the star invariants.

Proof. Let $\vec{u} = (u_1, \ldots, u_N)$ be the unit normal to the plane of the star. Then for each $j = 1, \ldots, N$, we have

$$\sum_{\ell=1}^n x_{\ell j} = \sqrt{n(\sigma + \rho)}\, u_j \,.$$

Thus

$$
\begin{aligned}
\left(\sum_{\ell=1}^n z_\ell \right)^2 &= \sum_{k,\ell=1}^n (x_{k1}x_{\ell 1} - x_{k2}x_{\ell 2} + 2ix_{k1}x_{\ell 2}) \\
&= n(\sigma + \rho)(u_1^2 - u_2^2 + 2iu_1 u_2) = n(\sigma + \rho)(u_1 + iu_2)^2 \,,
\end{aligned}
$$

whereas

$$\sum_{\ell=1}^n z_\ell^2 = \sum_{\ell=1}^n (x_{\ell 1}^2 - x_{\ell 2}^2 + 2ix_{\ell 1}x_{\ell 2}) = (WW^t)_{11} - (WW^t)_{22} + 2i(WW^t)_{12} = \sigma(u_1 + iu_2)^2 \,.$$

The formula now follows. $\qquad\square$

To test whether a framework in Euclidean space is invariant, it suffices to see whether the star about each of its vertices is invariant at the internal vertex. A consequence of the above corollary is that any *configured* star is invariant at its internal vertex. The star framework about the vertex of any regular polytope is configured, so that the underlying framework of a regular polytope is invariant [5]. On the other hand, not all invariant stars are configured. For example, the star in $\mathbb{R}^3$ with $2r$ external vertices represented by the columns of the $(3 \times (2r))$ –matrix

$$W = \begin{pmatrix} x_1 & x_2 & \cdots & x_r & x_1 & x_2 & \cdots & x_r \\ s_1 & s_2 & \cdots & s_r & -s_1 & -s_2 & \cdots & -s_r \\ t_1 & t_2 & \cdots & t_r & -t_1 & -t_2 & \cdots & -t_r \end{pmatrix} \,,$$

where the vectors $\vec{s} = (s_1, \ldots, s_r)$ and $\vec{t} = (t_1, \ldots, t_r)$ are orthogonal and of the same length, is invariant, but it is only configured when $x_1 = x_2 = \cdots = x_r$. This kind of invariant star arises in the double cone construction of the next section.

Let ϕ be a solution to (2.2) and consider a particular vertex x. Let $y_1, \ldots, y_n$ be the neighbours of x labelled in any order. Normalize the solution so that $\phi(x) = 0$ and then set $z_j = \phi(y_j)$. We suppose that not all z_j are zero. Write $z_j = \alpha_j + i\beta_j$ in real and imaginary parts. The lifting problem into $\mathbb{R}^3$ about the vertex x means finding an invariant star (whose internal vertex is located at the origin) with matrix

$$W = \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \beta_1 & \beta_2 & \cdots & \beta_n \\ x_1 & x_2 & \cdots & x_n \end{pmatrix}$$

If we impose the further restriction that the invariant star be configured, then provided $\gamma(x) < 1$ and rank $W = 3$, this can always be done with just a 2-fold ambiguity which corresponds to a choice of sign for the vector $\vec{x} := (x_1, \ldots, x_n)$ [5, 3]. The 2-fold ambiguity is illustrated by the Necker cube discussed in the Introduction. In the case when rank $W < 3$, then there is a 1-parameter family of solutions. This case occurs if and only if the complex numbers z_ℓ satisfy

$$n \sum_{\ell=1}^n |z_\ell|^2 + (\gamma - 2) \left| \sum_{\ell=1}^n z_\ell \right|^2 = 0 \,.$$

We may lift to an invariant configured star in $\mathbb{R}^N$ provided $N \leq n$, but for $N > 3$, there is generally a family of lifts [5].

Thus we find, that apart from special cases, we have a lift about each vertex of a solution to (2.2) into $\mathbb{R}^3$ which is unique modulo translation along the axis of projection $\mathbb{R}^3 \to \mathbb{R}^2$ and up to the 2-fold ambiguity corresponding to the sign of $\vec{x}$. This already enables certain geometric quantities to be defined in an unambiguous way, for example edge length. Note that for a given edge we may have liftings at each of its end points which endow that edge with different lengths. In that case, we can take the average as a reasonable definition of edge length. Furthermore, the 2-fold ambiguity may sometimes be removed by a requirement of global consistency, as is the case with the cube: a choice at one vertex imposes a choice of lift at neighbouring vertices.

The problem of when a global lifting of a given graph exists remains relatively unexplored. An obvious geometric obstruction occurs when, as discussed in the above paragraph, the lifts of neighbouring vertices defines a *different* length to the connecting edge. This is particularly relevant when we try to lift into $\mathbb{R}^3$ since then, in general, edge length is unique (see, for example [5], Example 4.4). However, in general there is a smooth family of lifts into $\mathbb{R}^N$ when $N > 3$ subject to the constraint that $N \leq n$ (n = degree of the vertex), so that it may still be possible to find a global lift into a higher dimension Euclidean space.

4. Examples

Given the interpretation of the parametre γ in terms of mean curvature, we now consider the problem of constructing invariant frameworks which have constant mean curvature. The underlying frameworks of regular polytopes provides examples, but are there others? The answer to this question is yes. Examples were given in [2], which we now outline.

If we take a regular n-sided polygon in the plane with vertices located at the points $e^{2k\pi i/n}$ ($k = 0, 1, \ldots, n-1$) and construct a double cone as illustrated below, then there is a unique height given by $\sin(2\pi/n)$ which makes this invariant, where by *height* we mean the distance from the plane of the polygon to one of the apexes.
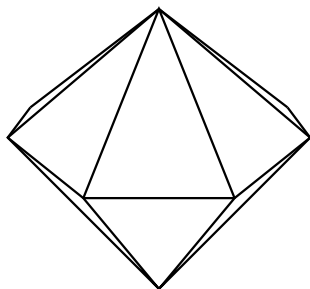
It is interesting to note that as $n \to \infty$ then the height approaches zero, so the form of the object approaches that of a disc. The value of γ at one of the lateral vertices is given by

$$(4.1) \qquad \gamma_{\text{lat}} = \frac{2(1 - 2\cos\frac{2\pi}{n} + 2\cos^2\frac{2\pi}{n})}{(2 - \cos\frac{2\pi}{n})^2},$$

whereas at one of the apexes it is given by

$$\gamma_{\text{apex}} = \frac{2\sin^2\frac{2\pi}{n} - 1}{2\sin^2\frac{2\pi}{n}}.$$

Invariance at either of the apexes is a consequence of Lemma 2. On the other hand, invariance at one of the lateral vertices, that is one of the vertices of the planar polygon, now connected to the two apexes as well as to its two polygonal neighbours, is far less obvious. The double cone only has constant mean curvature when $n = 4$, in which case it corresponds to the octahedron. However, we can remedy this by attaching another double cone along the axis of the two apexes as illustrated below, where, for convenience, we draw the cone axis horizontally. In essence, we adjust the length of the edge joining the double cones until the parameter γ coincides with its value at one of the lateral vertices given by (4.1).
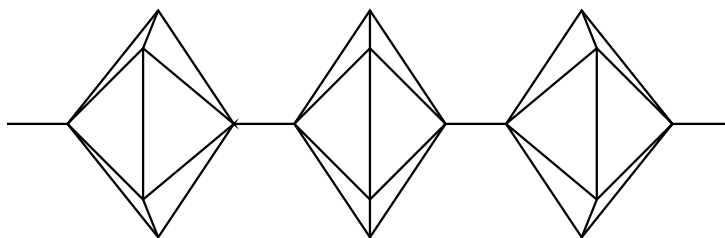
Such constant mean curvature frameworks are reminiscent of the period constant mean curvature surfaces of Delaunay [8].

If we let x denote the distance along the axis at which we must attach the next cone in order that the new value of γ at an apex coincides with the lateral value of γ, then x is determined by a quadratic equation. In the case when $n = 3$, one solution is given by $x = -\sqrt{3}$ which is precisely the distance between the two apexes, so we obtain the constant mean curvature framework illustrated in the left-hand figure below, corresponding to the complete graph on 5 vertices. The value of γ is given by $4/5$.
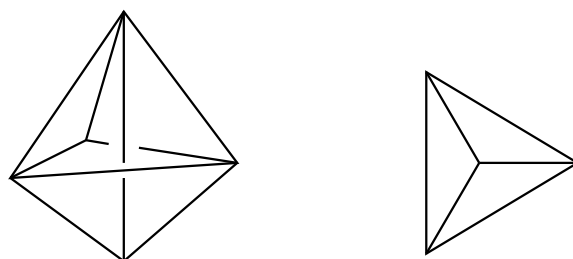
More generally, one can embed the complete graph on $N + 1$ vertices in an invariant way in $\mathbb{R}^{N-1}$ with γ constant equal to $N/(N + 1)$. The case when $N = 3$, corresponding to an embedding of the complete graph on 4 vertices in the plane, is illustrated in the right-hand figure above. This example illustrates the need to allow equivalence classes of functions in our definition of the geometric spectrum. Here, both $\Delta\phi$ and $(\nabla\phi)^2$ vanish at the central vertex, so that γ is not well-defined by equation (2.2) at this vertex. However, we can assign to it the value it takes at the other vertices.

For $n = 4$ and $n = 5$, there are no real solutions for x (as we would expect for $n = 4$ since the octahedron is already of constant mean curvature and the attachment of another cone would destroy this). For $n = 6$, there is both a positive and negative root. If we take the negative value and perform a twist and proceed similarly for successive cones, then we obtain interlaced frameworks with quite complicated structure.

As a final construction, we can take a tiling of the plane, form double cones and stack layers one upon the other to obtain an analogue of triply periodic constant mean curvatures [1]. Such a tiling by dodecahedrons and triangles is illustrated below.
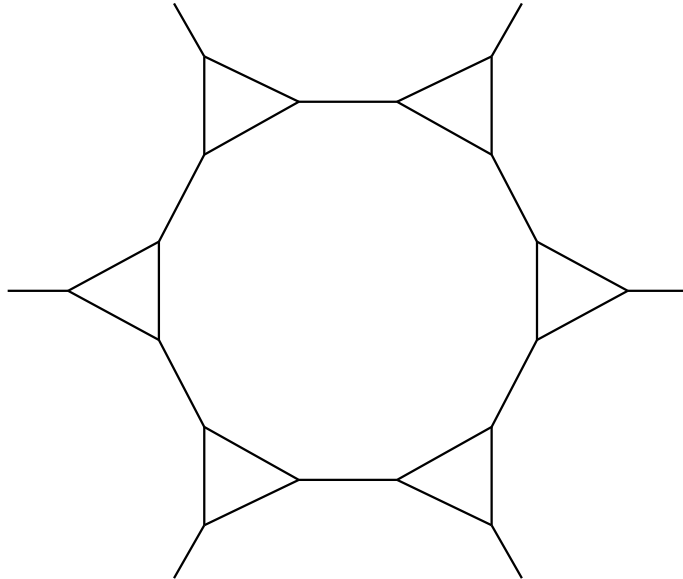
In this case, we form double cones over each triangle. Since the edges which connect each triangle (the edges of the dodecahedron) bisect the external angle, invariance is guaranteed. We then have to adjust the height of successive layers in order to arrange for γ to be constant.

The frameworks so constructed do not form the 1-skeleton of a polytope whose underlying topology is that of an immersed surface. We do not know if such examples exist other than the regular polytopes.

5. Other curvatures

By analogy with the smooth case, we view the function γ as related to mean curvature, at least when the framework is close to a smooth surface. But what about other curvatures? Various ideas were proposed in the arXiv article [1] which we now outline.

Suppose we have a solution to equation (2.2) and a well-defined lifting to a configured star at each vertex. For a given edge $e = \overline{xy}$ connecting vertices x and y, let $\theta(e)$ be the angle between the axes of the configured stars over x and y, respectively. Then this is well-defined and independent of the choice of representative solution under the equivalence (2.3). If we now let $\ell(e)$ denote the length of the edge $\overline{xy}$ as discussed at the end of Section 3, then the radius of the best-fit circle to that edge is $r(e) = \ell(e)/\theta(e)$. The reciprocal $k(e) := 1/r(e) = \theta(e)/\ell(e)$ may be taken as an analogue of *normal curvature*. This now depends on edge length and in particular on the choice of representative solution to (2.2). We can now make an alternative definition of mean curvature at a vertex x as the mean of the normal curvatures of edges incident with x. We do not know if after scaling is taken into account, there is a way to relate this to the function γ.

In smooth Riemannian geometry, the sectional curvature of a plane spanned by two vectors is the Gaussian curvature of a geodesic surface determined by the plane. In the discrete context, we can therefore define the sectional curvature associated to two edges e and f incident with a vertex x as $\mathrm{Sec}\,(e, f) := k(e)k(f)$.

Perhaps the most satisfying curvature in Riemannian geometry is the Ricci curvature which is scale invariant. For a given unit vector X, $\mathrm{Ric}\,(X, X)$ is the sum $\sum_j \mathrm{Sec}\,(X, Y_j)$ of the sectional curvatures of planes spanned by X and a set of orthonormal vectors $\{Y_j\}$ perpendicular to X. On a graph, for a given edge $e = \overline{xy}$ incident with a vertex x, we therefore define the Ricci curvature by:

$$\mathrm{Ric}_x(e, e) := \sum_{z \sim x, z \neq y} \theta(\overline{xy})\theta(\overline{xz})\,.$$

There seems to be no sensible way to define $\mathrm{Ric}\,(e, f)$ for distinct edges e and f incident with x. In the smooth case, this is usually achieved by the polarization identity, but there is no way to define the sum of two edges in the discrete context. Our definition of Ricci curvature is independent of the choice of representative solution to (2.2) under the equivalence (2.3). However, it is conceivable

that for a given γ there may be two distinct classes of solutions, so it is not clear if the Ricci curvature depends only on γ.

Let us finally consider the Gaussian curvature. For a convex polyhedron in $\mathbb{R}^3$, the classical theorem of Descartes affirms that the sum of the angular deficits at each vertex is equal to 4π [9]. By angular deficit $\delta(x)$ at a vertex x, we mean 2π minus the sum of the internal angles at x of the faces which contain x. In our case, at each vertex, we have a lifting to a configured star, but we do not a priori have any underlying polytope. Thus in order to define the Gaussian curvature in terms of angular deficit, we require an ordering of the edges. One way to do this is to edge colour the graph, as discussed in [1]. However, we only expect to obtain approximate global theorems with our method for the following reason.

Suppose we have an invariant framework in $\mathbb{R}^3$ which projects to a solution of (2.2). When we perform a lift at each vertex to try to recover the original framework, we lift to a *configured* star, whereas the original star may not be configured. In essence, we sacrifice global lifting for unicity of lifting (up to 2-valuedness). As an example, consider the invariant double cone on the triangle discussed above. Now there is an underlying polytope and we can calculate angular deficit at each vertex in the traditional way. By the theorem of Descartes, the total angular deficit is 4π. However, let us calculate it by taking a lift to a configured star at each vertex as determined by the corresponding solution to (2.2). In the original figure, the stars at the lateral vertices are not configured, so an error will occur. We find:

$$\delta_{\text{apex}} = 2\pi - 3\arccos\frac{1}{7} \quad \text{and} \quad \delta_{\text{lat}} = 2\pi - 4\arccos\frac{5}{7},$$

to give a total curvature of

$$\delta_{\text{tot}} = 3\delta_{\text{lat}} + 2\delta_{\text{apex}} = 10\pi - 6\left(\arccos\frac{1}{7} + 2\arccos\frac{5}{7}\right) \sim 4.244 \times \pi.$$

References

[1] P. Baird. A class of quadratic difference equations on a finite graph. *arXiv:1109.3286 [math-ph]*.

[2] P. Baird. Constant mean curvature polytopes and hypersurfaces via projections. *Differential Geom. and its Applications*, online version nov. 2013.

[3] P. Baird. Emergence of geometry in a combinatorial universe. *J. Geom. and Phys.*, **74**:185–195, (2013).

[4] P. Baird. Information, universality and consciousness: a relational perspective. *Mind and Matter*, **11**(2):21–43, (2013).

[5] P. Baird. An invariance property for frameworks in euclidean space. *Linear Algebra and its Applications*, **440**:243–265, (2014).

[6] P. Baird and M. Wehbe. Twistor theory on a finite graph. *Comm. Math. Phys.*, **304**(2):499–511, (2011).

[7] S. Barré. Real and discrete holomorphy: introduction to an algebraic approach. *J. Math. Pures Appl.*, **87**:495–513, (2007).

[8] C. Delaunay. Sur la surface de révolution dont la courbure moyenne est constante. **6**:309–320, (1841).

[9] R. Descartes. Progymnasmata de solidorum elementis. *Oeuvres de Descartes*, X:265–276.

[10] M. G. Eastwood and R. Penrose. Drawing with complex numbers. *Math. Intelligencer*, **22**:8–13, (2000).

Laboratoire de Mathématiques, de Bretagne Atlantique, Université de Bretagne Occidentale, 6 av. Victor Le Gorgeu – CS 93837, 29238 BREST CEDEX, FRANCE • paul.baird@univ-brest.fr

Relaxations for Minimizing Metric Distortion and Elastic Energies for 3D Shape Matching

Daniel CREMERS, Emanuele RODOLÀ, and Thomas WINDHEUSER

Abstract

We present two methods for non-rigid shape matching. Both methods formulate shape matching as an energy minimization problem, where the energy measures distortion of the metric defined on the shapes in one case, or directly describes the physical deformation relating the two shapes in the other case. The first method considers a parametrized relaxation of the widely adopted quadratic assignment problem (QAP) formulation for minimum distortion correspondence between deformable shapes. In order to control the accuracy/sparsity trade-off a weighting parameter is introduced to combine two existing relaxations, namely spectral and game-theoretic. This leads to an approach for deformable shape matching with *controllable sparsity*. The second method focuses on computing a geometrically consistent and spatially dense matching between two $3D$ shapes. Rather than mapping points to points it matches infinitesimal surface patches while preserving the geometric structures. In this spirit, matchings are considered as diffeomorphisms between the objects' surfaces which are by definition geometrically consistent. Based on the observation that such diffeomorphisms can be represented as closed and continuous surfaces in the product space of the two shapes, this leads to a minimal surface problem in this product space. The proposed discrete formulation describes the search space with linear constraints. Computationally, the approach results in a binary linear program whose relaxed version can be solved efficiently in a globally optimal manner.

1. INTRODUCTION

An increasing number of digitized three-dimensional objects has become available over the last years due to the technical progress in acquisition hardware like laser scanners or medical imaging devices. Such objects originate from a variety of different domains including biology, medicine, industrial design or computer animation. This rapid growth in stored data brings about the need for reliable algorithms to organize this data. One of the cornerstone problems in this context is the matching problem: In its most typical form, it concerns the problem of determining a map $f : X \to Y$ among two given shapes in such a way that their geometrical properties are preserved by the transformation. A particularly challenging instance of this problem occurs when the two shapes undergo general non-rigid deformations. As such, matching of deformable shapes has attracted the interest of researchers over the years and a wide variety of approaches have been proposed (see, e.g. [3] and references therein for a recent comparison).

A prominent approach to the matching problem from a metric perspective was introduced in [17], a concept that was explored further in [4] with the introduction of the GMDS framework, where the minimum distortion isometric embedding of one surface onto another is explicitly sought. A different view on the problem stems from the notion of uniformization space [14, 32]. Lipman and Funkhouser [14] proposed to model deviations from isometry by a transportation distance between corresponding points in a canonical domain (the complex plane); the result of this process is a

"fuzzy" correspondence matrix, whose values can be given the natural interpretation of confidence levels attributed to each match. Fuzzy schemes are typically adopted to relax the point-to-point mappings [18, 21]. Lipman and Daubechies [13] proposed to compare surfaces of genus zero and open surfaces using optimal mass transport and conformal geometry. Computationally, this amounts to solving a linear program in n^2 variables where n is the number of vertices used in the discretization of the surfaces. The problem with this approach is that no spatial regularity is imposed on the matchings. In general, while methods based on uniformization theory are made attractive by the low dimensionality of the embedding domain, they do not behave well with different kinds of deformations (e.g., topological changes), and are subject to global inconsistencies in the final mapping.

In this work, we consider two different approaches to deformable shape matching. The two approaches share the common perspective of minimizing a distortion criterion, derived from the metric information which the shapes to be matched are endowed with. In one case (Section 3), following [23], we consider a notion of pairwise metric distortion that directly captures to what extent two shapes can be isometrically put in correspondence. Motivated by the observation that good correspondences often come at the price of high sparsity (in terms of number of matched points), whereas large cardinality tends to bring distorted matches into the correspondence, we attempt to control the accuracy/sparsity trade-off by introducing a weighting parameter on the combination of two effective relaxations [12, 21], which we relate to their regularizer counterparts from regression analysis. This leads us to the introduction of the *elastic net* penalty function [33] into shape matching problems. Differently, our second approach [30] takes a physically motivated view on the problem and minimizes a functional that encodes the physical deformation energy [15, 31] necessary to deform one shape into the other. The formulation we give in Section 4 is based on finding an optimal surface of codimension 2 in the product of the two shape surfaces. We derive a consistent discretization of the continuous framework and show that the discrete minimal surface problem amounts to a linear program. Compared to existing approaches, our construction involves the boundary operator [27, 10, 25], and guarantees a geometrically consistent matching in the sense that the surfaces are mapped into one another in a continuous and orientation preserving manner.

2. Energy functionals for measuring the matching quality

In this section we discuss the matching energies that have been used to find correspondences among shapes in [23] and [30] respectively.

2.1. **Minimum metric distortion.** We model shapes as compact Riemannian manifolds endowed with an intrinsic metric d. A point-to-point *correspondence* between two shapes X and Y is defined as a subset $C \subset X \times Y$ satisfying: 1) for every $x \in X$, there exists at least one $y \in Y$ such that $(x, y) \in C$, and vice versa, 2) for every $y \in Y$, there exists $x \in X$ such that $(x, y) \in C$. This relation can be alternatively formulated as a binary function $c : X \times Y \to \{0, 1\}$ satisfying the mapping constraints

$$(2.1) \qquad \max_{x \in X} c(x, y) = \max_{y \in Y} c(x, y) = 1,$$

for every $y \in Y$ and $x \in X$. According to this definition, clearly not all correspondences give rise to meaningful matches among the two given shapes (consider, for instance, the full Cartesian product given by $c(x, y) = 1$ for all $(x, y) \in X \times Y$). A common requirement in this setting is that the correspondence should represent a bijective mapping, or more typically an *isometry* between the two surfaces. With this requirement in mind, in order to give a measure of *quality* to the correspondence we evaluate the distortion induced by the mapping as measured on the two shapes using the respective metrics d_X and d_Y. In particular, given two matches $(x, y), (x', y') \in C$, the absolute criterion

$$(2.2) \qquad \epsilon(x, y, x', y') = |d_X(x, x') - d_Y(y, y')|$$

directly quantifies to what extent the estimated correspondence deviates from isometry. Following [18, 21], we first relax the correspondence from a discrete to a fuzzy notion by letting $c : X \times Y \to [0, 1]$, effectively setting off the problem from its combinatorial nature and bringing

it to a continuous optimization domain. Further, following a similar approach to the Gromov-Wasserstein [18] family of metrics, we obtain a *relaxed* notion of proximity between shapes:

$$(2.3) \qquad D(X,Y) = \frac{1}{2} \min_C \sum_{(x,y),(x',y') \in C} \epsilon^p(x,y,x',y') c(x,y) c(x',y').$$

Note from this definition that we don't require the two shapes to have a measure defined over them (differently from [18, 21]). Establishing a minimum distortion correspondence between the two shapes amounts to finding a minimizer of the above distance. To this end, the problem can be easily recast as a relaxed quadratic assignment problem (QAP) [16],

$$(2.4) \qquad \min_{\mathbf{C}} \quad \mathrm{vec}\{\mathbf{C}\}^T \mathbf{A} \, \mathrm{vec}\{\mathbf{C}\}$$
$$\text{s.t.} \quad \mathbf{C1} = 1, \ \mathbf{C}^T \mathbf{1} = 1, \ \mathbf{C} \succeq 0,$$

where $\mathrm{vec}\{\mathbf{C}\}$ is the $|C|$-dimensional column-stack vector representation of the correspondence matrix $\mathbf{C}$, $\mathbf{A}$ is a non-negative symmetric cost matrix containing the pairwise distortion terms that appear in (2.3), $\mathbf{1}$ is a vector of $n = |C|$ ones, and $\succeq$ denotes element-wise inequality. 1 We emphasize that, although easier to solve, the relaxation provided above is still non-convex. Note that in the standard QAP, function c is taken to be a binary correspondence and matrix $\mathbf{C}$ is thus required to be a permutation matrix. The QAP is a NP-hard problem due to the combinatorial complexity of this latter constraint.

2.2. **Elastic deformation energies.** A different approach to model a matching energy between shapes is to restrict the class of deformations that transform one shape into another to the set of diffeomorphisms. This gives us two benefits. First, the shapes do not get "cut" open during the matching transformation. Second, we can assign to each diffeomorphism an elastic energy that directly gives us a physical interpretation of the matching.

In the following, we assume that the two shapes $X, Y \subset \mathbb{R}^3$ are differentiable, oriented, closed surfaces. Diffeomorphisms $f : X \to Y$ are bijections for which both f and f^{-1} are differentiable. We formulate the shape matching problem as an optimization problem over the set of orientation preserving diffeomorphisms between X and Y,

$$(2.5) \qquad \inf_{f \in \mathrm{Diff}^+(X,Y)} E(f) + E(f^{-1})$$

where E is a suitable energy on the class of all diffeomorphisms between surfaces and $\mathrm{Diff}^+(X,Y)$ is the set of orientation preserving diffeomorphisms between X and Y. Note that we choose a symmetric problem formulation, penalizing at the same time deformation energy of X into Y and of Y into X. This is necessary because usually E takes different values on f and on f^{-1}.

The energy functional we use is borrowed from elasticity theory in physics [5], which interprets the surfaces X and Y as "thin shells". Now we try to find the deformation of X into Y which requires the least stretching and bending energy. Such models usually consist of a membrane energy E_{mem} and a bending energy E_{bend} penalizing deformations in the first and in the second fundamental forms of the surfaces. In this work we use the following formulation:

$$(2.6) \qquad E(f) = \underbrace{\int_X (\mathrm{tr}_{g_X} \mathbf{E})^2 + \mu \, \mathrm{tr}_{g_X}(\mathbf{E}^2)}_{E_{\mathrm{mem}}} + \lambda \underbrace{\int_X (H_X(x) - H_Y(f(x)))^2}_{E_{\mathrm{bend}}}$$

where $\mathbf{E} = f^* g_Y - g_X$ is the difference between the metric tensors of X and Y, typically called the *Lagrange strain tensor*, $\mathrm{tr}_{g_X}(\mathbf{E})$ is the norm of this tensor (see [8]), H_X and H_Y denote the mean curvatures and μ and λ are parameters which determine the elasticity and the bending property of the material. This energy is a slightly simplified version of Koiter's thin shell energy [11].

3. Minimum distortion correspondence via Elastic Net regularization

In this Section we present three different relaxations to the minimal metric distortion as formulated in problem (2.4). The three approaches act by relaxing the mapping constraints imposed on the correspondence function $c(x,y)$. Even though originating from distinct motivations, the first two methods share a convenient interpretation as partitioning problems in the space of potential assignments. In Section 3.3 we provide a different view on the problem, as presented in [23, 22], by using the language of regression analysis.

3.1. **Spectral matching.** Taking the point of view of graph clustering, [12] proposed the simplified problem

$$(3.1) \qquad \min_{\mathbf{x}} \ \mathbf{x}^T \mathbf{A} \mathbf{x} \quad \text{s.t.} \|\mathbf{x}\|_2^2 = 1 \,,$$

where $\mathbf{x} \equiv \text{vec}\{\mathbf{C}\} \in \mathbb{R}^n$ is the vector representation for the correspondence. Following Rayleigh's quotient theorem, this modified QAP is minimized by the eigenvector $\mathbf{x}^\star$ corresponding to the minimum eigenvalue of $\mathbf{A}$. Note that mapping constraints are not imposed in (3.1). The authors follow a greedy algorithm to impose such constraints only after a solution has been obtained. The method has a tendency to produce matches for each point. Nevertheless, symmetries and structured noise in the data (indeed a characteristic of the non-rigid setting) may lead to unstable eigenvectors [12] and thus unreliable assignments.

A useful interpretation to this approach can be given as a relaxed two-way partitioning problem [1]. Consider the set of constraints taking the form $\mathbf{x}_i^2 = 1$ for $i = 1 \ldots n$; these constraints restrict the values of $\mathbf{x}_i$ to ± 1, so the problem is equivalent to finding the partitioning (as "match" or "non-match") on a set of n elements that minimizes the total cost $\mathbf{x}^T \mathbf{A} \mathbf{x}$. Here, the coefficients $\mathbf{A}_{ij}$ can be interpreted as the cost of having elements i and j in the same partition. Clearly, the new constraints imply $\sum_{i=1}^n \mathbf{x}_i^2 = \|\mathbf{x}\|_2^2 = n$; since this actually allows the $\mathbf{x}_i$ to take on any (small enough) real number, optimizing over this feasible set will yield a lower bound on the optimal value of the original partitioning problem.

3.2. **Game-theoretic matching.** Given the inherent difficulty to solve for a minimum distortion correspondence under general deformations, we recently proposed to shift the focus to the search of a group of matches having least distortion, *regardless* of its cardinality [21]. To achieve this, we proposed to optimize over the probability simplex

$$(3.2) \qquad \|\mathbf{x}\|_1 = \mathbf{1}^T \mathbf{C} \mathbf{1} = 1 \,, \ \mathbf{x} \succeq 0 \,.$$

In this formulation, the space of assignments is in a one-to-one correspondence with all possible probability distributions of a random variable, realizing as $\mathbf{x}$, modeling the concept of match. The main benefits of adopting such L^1-type constraint for the matching problem arise from its convenient game-theoretical interpretation, leading to very efficient algorithms for (local) optimization and, most remarkably, in allowing the mapping constraints to be embedded directly into the cost matrix $\mathbf{A}$. Unfortunately, the strong locality and selectivity demonstrated by the game-theoretic approach is hardly desirable for matching problems.

Similarly to the L^2 case, the game-theoretic approach can be regarded as an attempt to solve a partitioning problem where the two partitions are represented by $\mathbf{x}_i = 0$ or 1 for $i = 1 \ldots n$. This, in turn, corresponds to imposing a bound on the "counting" norm $\|\mathbf{x}\|_0$, which is relaxed here to the continuous sparsity-inducing counterpart $\sum_{i=1}^n |\mathbf{x}_i| = \|\mathbf{x}\|_1 = n$, with $\mathbf{x}_i \geq 0$ for all i.

3.3. **Matching with the Elastic Net.** In practical settings, the performance of the framework given in Section 2.1 directly depends on the definition of the metric distortion term ϵ. This is, in fact, a property shared by *any* method attempting to minimize (2.4). Ovsjanikov et al. [20] recently introduced the notion of shape condition number. According to this notion, the stability of the matching can be characterized as an intrinsic property of the shape itself, and is related to its intrinsic symmetries as well as the specific choice of a metric.

In order to incorporate a somewhat elusive notion of stability into the matching process, we propose to change the point of view by drawing an analogy between the correspondence problem and model-fitting. Our goal, in this context, is to determine a good approximation of the true relationship between the two shapes: we seek to *fit* or approximate the optimal correspondence $\mathbf{x}^\star$ as closely as possible, with deviation measured in the quadratic form $\mathbf{x}^T \mathbf{A} \mathbf{x}$. Problems of this kind are often studied with the tools of regression analysis [1]. Here the interest shifts from finding a best fit to analyzing the relationships among the several variables that build up the set of potential assignments $\{\mathbf{x}_i\}_{i=1 \ldots n}$. These candidate matches act as predictors for the minimum distortion correspondence, and can be given the interpretation of explanatory variables which we observe, while we seek to find the combination that best describes the data in the minimal distortion sense. Since in general these variables hold a certain degree of correlation among them, it is of particular interest to attempt to determine whole groups of highly correlated predictors, as they will likely form consistent groups of matches in terms of the adopted measure of distortion.
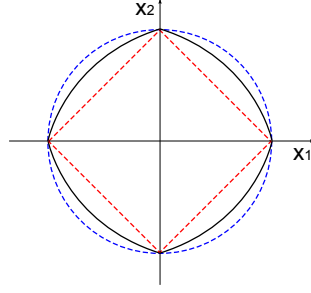
Figure 3.1: Contour plots of the L^2 (circle), L^1 (diamond), and elastic net (in between) balls in $\mathbb{R}^2$. In this example we set $\alpha = 0.6$. The strength of convexity varies with α.

In this view, spectral matching can be directly related to ridge regression, whereas the game-theoretic technique finds its equivalent in the lasso, the sparsity-inducing L^1 regularizer performing continuous shrinkage and automatic variable selection simultaneously [1, 33]; one major limitation of the lasso is its tendency to select only one variable from a group of variables among which the pairwise correlations are very high. In order to strike a balance between the two methods, we adopt a family of constraints known as elastic net [33]. This regularization technique shares with the lasso the ideal property of performing automatic variable selection, and most notably it is able to select entire groups of highly correlated variables. The elastic net criterion is defined as a convex combination of the lasso and ridge penalties:

$$(3.3) \qquad (1-\alpha)\|\mathbf{x}\|_1 + \alpha\|\mathbf{x}\|_2^2\,,\ \alpha \in [0,1]\,.$$

This penalty function is singular at 0 and *strictly* convex for $\alpha > 0$, thus possessing the characteristics of both penalties (see Fig. 3.1). Strict convexity plays an important role as it guarantees the grouping effect in the extreme situation with identical predictors (that is, whenever the distortion between two matches is exactly 0), and provides a quantitative description of their degree of correlation otherwise. Let $\mathbf{x} \in \mathbb{R}^{|C|}$ be the vector representation of some correspondence $C \subset X \times Y$, we expect the elastic net-penalized solution to keep the difference $|\mathbf{x}_i - \mathbf{x}_j|$ small whenever the metric distortion $\epsilon(C_i, C_j)$ between the two matches is small. The trade-off between size of the correspondence and matching error is regulated by the convexity parameter α, which allows to fine tune the model complexity and balance the action of the penalty ranging from the highly selective pure lasso for $\alpha = 0$ to the more tolerant ridge behavior for $\alpha = 1$. This leads to the following family of relaxations for the QAP:

$$(3.4) \qquad \begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{x}^T \mathbf{A} \mathbf{x} \\ \text{s.t.} \quad & (1-\alpha)\|\mathbf{x}\|_1 + \alpha\|\mathbf{x}\|_2^2 = 1,\ \mathbf{x} \succeq 0\,, \end{aligned}$$

with $\alpha \in [0,1]$. The family directly generalizes the spectral and game-theoretic techniques. Similarly to the spectral approach, this formulation does not guarantee the final solution to represent a bijective mapping, which can nevertheless be efficiently obtained *a posteriori* as in [12].

3.3.1. *Optimization.* We undertake a projected gradient approach [1] to determine a local optimum for problem (3.4). The optimization process is governed by the equations

$$(3.5) \qquad \mathbf{x}^{(t+1)} = \Pi_\alpha\left(\mathbf{x}^{(t)} - \gamma^{(t)}\mathbf{A}\mathbf{x}^{(t)}\right),$$

where $\mathbf{A}\mathbf{x} = \frac{1}{2}\nabla\mathbf{x}^T\mathbf{A}\mathbf{x}$ is a descent direction for the objective, $\gamma > 0$ is the step length taken in that direction, and $\Pi_\alpha : \mathbb{R}^n \to \mathbb{R}^n$ is a projection operator taking a solution back onto the feasible set. We initialize $\mathbf{x}^{(0)}$ to the barycenter of the elastic net boundary, i.e., for all $i = 1\dots n$ we set $\mathbf{x}_i$ to the positive solution of the quadratic equation $\alpha n x^2 + (1-\alpha)n x - 1 = 0$.

While efficient methods for projecting onto the L^2 and L^1 balls have been proposed in literature [26], projection onto their convex combination is a more involved task. Computing the Euclidean projection $\Pi_\alpha(\mathbf{x}_0)$ onto the (positive) elastic net ball boundary amounts to solving the
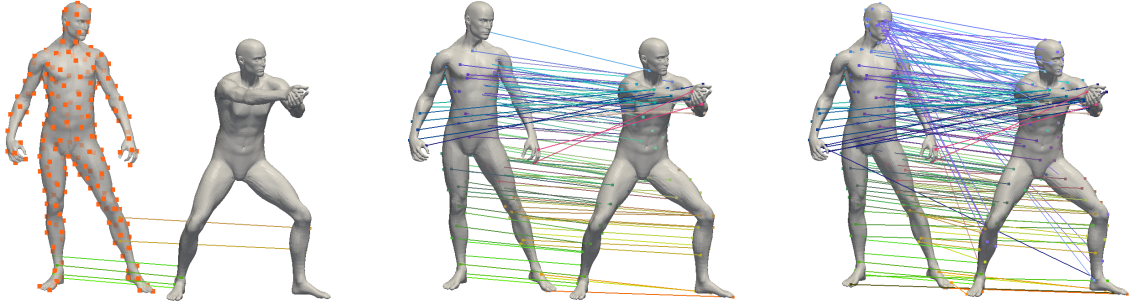
Figure 3.2: Example of matchings obtained with the game-theoretic, elastic net and spectral techniques respectively. See Section 3.3.2 for details.

following problem:

$$(3.6) \qquad \min_{\mathbf{x}} \quad \|\mathbf{x} - \mathbf{x}_0\|_2^2$$
$$\text{s.t.} \quad (1-\alpha)\mathbf{1}^T\mathbf{x} + \alpha\mathbf{x}^T\mathbf{x} = t, \ \mathbf{x} \succeq 0 \,,$$

with $\alpha \in [0,1]$. A detailed explanation of our approach on the computation of the unique minimum-distance projection Π_α in an efficient manner is given in [23]. Also note that, for practical purposes, we adopt a more efficient alternative to the standard projected gradient descent (3.5), namely its acceleration via vector extrapolation techniques [22].

3.3.2. *Experimental results.* We performed a wide range of experiments on the SHREC'10 standard dataset [3], which includes shapes undergoing several different types of deformation, e.g. quasi-isometric deformations, topological changes, displacement noise and changes in scale (we refer to [23] for a detailed numerical breakdown). Differently from most existing methods, the approach presented in this Section is quite general and not restricted to the quasi-isometric case. Indeed, invariance to different kinds of deformations is induced by the proper choice of the metrics employed in (2.2) (see [21] for an example). In order to make the computational task more tractable, only a limited number of samples are considered from one shape, and then potential matches are built with the 5 points from the other shape having similar curvature. Samples are generated via farthest point sampling (FPS) [17, 18] using the extrinsic Euclidean metric, a technique allowing to cover the whole surface in a sparse manner while retaining the metric information contained in the initial shape as best as possible. Note that only one of the two shapes is subsampled, while we keep all points in the other.

Fig. 3.2 presents an example in which the correct matches have a very small inlier ratio with respect to the set of candidates. In this matching scenario, our method provides a means to select only high-precision correspondences in a situation where there is huge ambiguity in most correspondences. In this example, the set of potential assignments is constructed by taking $\sim$200 farthest points on one shape, and then building the whole Cartesian product with the *correct* corresponding points from the other shape, after 45% of them have been moved to random positions over the surface. This setup simulates a moderately challenging scenario in which only $\sim$50% of the shape is matchable with low distortion, and the feasible set comprises all possible assignments between the two shapes. The game-theoretic (L^1) solution is highly selective and only assigns 3% of the shape samples accurately (left image); in contrast, the spectral (L^2) approach favors dense solutions and yields matches for 93% of the points with large error (right image). Elastic net matching (middle) allows to regulate the trade-off between size and distortion: the correspondence is made more dense, and 53% of the points are matched while keeping the error small. Here we set $\alpha = 0.85$.

4. Minimizing the elastic energy via linear programming relaxation

In this section we will discuss the approach presented in [30] that tries to solve the elastic energy problem

$$(4.1) \qquad \inf_{f \in \text{Diff}^+(X,Y)} E(f) + E(f^{-1})$$

already introduced in Section 2.2. The approach puts the focus on three aspects:
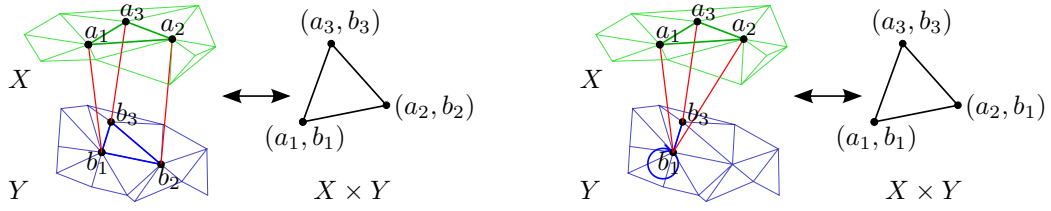
Figure 4.1: The construction of product triangles that make up the discrete version of the product space $X \times Y$. **Left image:** The triangle $(a_1, a_2, a_3)^T$ on surface X is matched to triangle $(b_1, b_2, b_3)^T$ on Y by assigning vertex a_i to vertex b_i. This directly corresponds to the triangle with vertices (a_i, b_i) in the product graph. **Right image:** The triangle $(a_1, a_2, a_3)^T$ is matched to the edge $(b_1, b_3)^T$, represented here as degenerate triangle $(b_1, b_1, b_3)^T$.

(1) Representation of the set of orientation-preserving diffeomorphisms $\mathrm{Diff}^+(X, Y)$,
(2) discretization of this set and the energy E, and
(3) optimization of the discrete version of the energy.

The main idea underlying our representation is to look at subsets of the product space $X \times Y$. We will introduce constraints such that these subsets become graphs of diffeomorphisms. We will show further how we can discretize the product space, the constraints and the energy. Interestingly, the constraints and the energy are linear in the variables that span the discretized version of $X \times Y$. The resulting optimization problem is thus an *integer linear program* (ILP).

While we cannot find the global optimum of this optimization problem we can allow non-integer solutions and transform the ILP into a *linear program* (LP). The global optimum of the linear program can be computed in polynomial time and is a lower bound of the original optimization problem.

4.1. **Diffeomorphisms and their graph surfaces.** Given an orientation preserving diffeomorphism $f : X \to Y$ we obtain a set $\Gamma \subset X \times Y$ in the Euclidean product of X and Y by passing to the graph

$$(4.2) \qquad \Gamma = \{(x, f(x)) \mid x \in X\} \subset X \times Y.$$

The set Γ comes with two natural projections $\pi_X : \Gamma \to X, (x, f(x)) \mapsto x$ and $\pi_Y : \Gamma \to Y, (x, f(x)) \mapsto f(x)$. A diffeomorphism is completely characterized by its graph:

Proposition 1 (graph surfaces). *Let Γ be the graph of a diffeomorphism $f : X \to Y$. Then*

(i) *Γ is a differentiable, connected, closed surface in the product space $X \times Y$.*
(ii) *The projections π_X and π_Y are both diffeomorphisms.*
(iii) *The two orientations which Γ naturally inherits from X and Y coincide.*

Conversely, any subset $\Gamma \subset X \times Y$ which satisfies (i),(ii) and (iii) is the graph of an orientation-preserving diffeomorphism between X and Y. We call such sets graph surfaces.

The energy $E(f)$ can be expressed as

$$(4.3) \qquad E(f) = \tilde{E}(\Gamma)$$

where $\tilde{E}(\Gamma) = E(\pi_Y \circ (\pi_X)^{-1}) + E(\pi_X \circ (\pi_Y)^{-1})$.

The outcome of the above discussion is that the optimization problem (4.1) can be phrased as an optimization problem over the set of subsets of $X \times Y$, which then reads

$$(4.4) \qquad \begin{array}{ll} \inf & \tilde{E}(\Gamma) \\ \text{subject to} & \Gamma \subset X \times Y \text{ is a graph surface} \end{array}$$

We remark that the idea of casting optimal diffeomorphism problems as minimal surface problems has been applied previously in the theory of nonlinear elasticity [9]. In the setup of shape matching, it is related to the approach that Tagare [28] proposed for the matching of 2D shapes. It was reformulated as an orientation preserving diffeomorphism approach in [24].
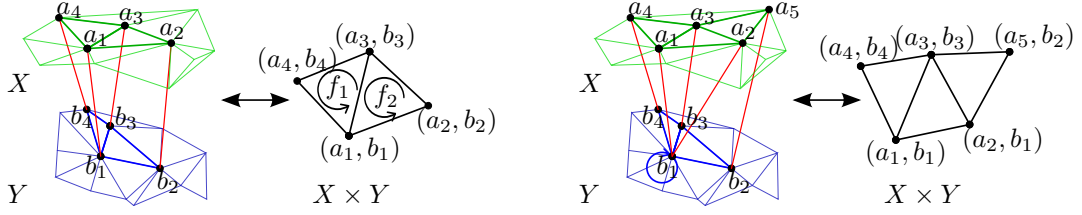
Figure 4.2: The discrete version of condition (i) includes the closeness condition ensuring that neighboring triangles on X are matched with neighboring triangles on Y. **Left image (general case):** The triangles $(a_1, a_2, a_3)^T$ and $(b_1, b_2, b_3)^T$ are matched resulting in activating f_2. The boundary condition $\partial \Gamma = 0$ ensures that the matching continues with a correspondence whose triangles in X and Y are positively incident to $(a_1, a_3)^T$ and $(b_1, b_3)^T$ respectively. This constraint is satisfied for example by triangle f_1 which is visualized here. **Right image (stretching):** The stretching is achieved by matching triangle $(a_1, a_2, a_3)^T$ to edge $(b_3, b_1)^T$. Again, the closeness condition is granted by the boundary operator evaluated on the product edges $((a_2, b_1), (a_3, b_3))^T$ and $((a_3, b_3), (a_1, b_1))^T$.

4.2. **The discrete setting.** We develop now a discrete counterpart of the notion of graph surfaces in $X \times Y$ and the continuous elastic matching energy by assuming that the surfaces X, Y are given as triangulated meshes.

4.2.1. *Discrete surface patches.* Let $X = (V_X, E_X, F_X)$ be a triangulated oriented surface mesh, consisting of a set of vertices V_X, of directed edges E_X and of oriented triangles F_X. A priori, edges on X do not have a preferable orientation. Therefore, we fix an orientation for each edge on X. Thus, whenever two vertices a_1 and a_2 of X are connected by an edge, either $\left(\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right) \in E_X$ or $\left(\begin{smallmatrix} a_2 \\ a_1 \end{smallmatrix}\right) = -\left(\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right) \in E_X$. We extend the set of edges by *degenerate edges* $\overline{E}_X = E_X \cup \{\left(\begin{smallmatrix} a \\ a \end{smallmatrix}\right) \mid a \in V_X\}$. By assumption, the triangular faces of X are oriented. If the vertices a_1, a_2, a_3 build an oriented triangle on X, then $\left(\begin{smallmatrix} a_1 \\ a_2 \\ a_3 \end{smallmatrix}\right) = \left(\begin{smallmatrix} a_2 \\ a_3 \\ a_1 \end{smallmatrix}\right) = \left(\begin{smallmatrix} a_3 \\ a_1 \\ a_2 \end{smallmatrix}\right) \in F_X$. Similarly, we extend the set of triangles by *degenerate triangles* $\overline{F}_X = F_X \cup \left\{ \left(\begin{smallmatrix} a_1 \\ a_2 \\ a_2 \end{smallmatrix}\right) \mid a_1, a_2 \in V_X, \pm \left(\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right) \in \overline{E}_X \right\}$. Notice that degenerate triangles can consist of only one or two vertices. The existence of these degenerate triangles will allow stretching or compression of parts of the surface.

Next, we introduce product triangles for two triangular meshes X and Y. Define the product of X and Y by the set of vertices $V = V_X \times V_Y$, the set of edges $E = \overline{E}_X \times \overline{E}_Y$ and the set of product triangles

$$(4.5) \qquad F := \left\{ \begin{pmatrix} (a_1, b_1) \\ (a_2, b_2) \\ (a_3, b_3) \end{pmatrix} \middle| \begin{array}{c} f_1 = \left(\begin{smallmatrix} a_1 \\ a_2 \\ a_3 \end{smallmatrix}\right) \in \overline{F}_X, \\ f_2 = \left(\begin{smallmatrix} b_1 \\ b_2 \\ b_3 \end{smallmatrix}\right) \in \overline{F}_Y, \\ f_1 \text{ or } f_2 \text{ non-degenerate} \end{array} \right\}$$

We will call the triple (V, E, F) the *product graph* as the discrete counterpart of the product space $X \times Y$. The product triangles in F are the basic pieces which are later glued to discrete graph surfaces. For shape matching, a product triangle $((a_1, b_1), (a_2, b_2), (a_3, b_3)) \in F$ is interpreted as setting vertex $a_i \in V_X$ in correspondence with vertex $b_i \in V_Y$ (see Figure 4.1).

4.2.2. *Discrete surfaces.* Following Proposition 1 a diffeomorphism can be represented as a surface $\Gamma \subset X \times Y$ satisfying conditions (i), (ii) and (iii). In this section we derive discrete versions of these properties.

Definition 2. A *discrete surface* in $X \times Y$ is a subset $\Gamma \subset F$. The set of all discrete surfaces is denoted by surf$(X \times Y)$.

As we have seen above, a product triangle in F can be interpreted as matching a triangle on X to a triangle on Y. Thus, the intuitive meaning of a discrete surface $\Gamma \subset F$ is a set of correspondences

between triangles on X and Y. Imposing the discrete counterparts of (i), (ii) and (iii) on such a discrete surface will result in the discrete counterpart of a diffeomorphic matching.

Discrete version of (i): In the following we will find a condition which guarantees the continuity of our matching. Recall that the boundary operator for triangle meshes [7] maps triangles to their oriented boundary. We extend this definition to the product graph G.

As for the sets E_X and E_Y we choose arbitrary orientations for each product edge $e \in E$. We then define for two vertices $v_1, v_2 \in V$ a vector $O\left(\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right) \in \mathbb{Z}^{|E|}$ whose e-th entry is given by

$$(4.6) \qquad O\left(\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right)_e = \begin{cases} 1 & \text{if } e = \left(\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right) \\ -1 & \text{if } e = \left(\begin{smallmatrix} v_2 \\ v_1 \end{smallmatrix}\right) \\ 0 & \text{else.} \end{cases}$$

The triangles in F naturally inherit orientations from the triangles in F_X and F_Y. This allows us to define the boundary operator as follows.

Definition 3. The *boundary operator* $\partial : F \to \mathbb{Z}^{|E|}$ is defined by

$$(4.7) \qquad \partial \begin{pmatrix} a_1, b_1 \\ a_2, b_2 \\ a_3, b_3 \end{pmatrix} := O \begin{pmatrix} a_1, b_1 \\ a_2, b_2 \end{pmatrix} + O \begin{pmatrix} a_2, b_2 \\ a_3, b_3 \end{pmatrix} + O \begin{pmatrix} a_3, b_3 \\ a_1, b_1 \end{pmatrix},$$

where the $a_i \in V_X$ and $b_i \in V_Y$ form triangles on X resp. on Y and $\left(\begin{smallmatrix} a_i, b_i \\ a_j, b_j \end{smallmatrix}\right)$ is the product edge connecting the product vertices (a_i, b_i) and (a_j, b_j). The boundary operator is linearly extended to a map $\partial : \mathrm{surf}(X \times Y) \to \mathbb{Z}^{|E|}$. A discrete surface Γ in $X \times Y$ is *closed* if $\partial \Gamma = 0$.

The closeness condition ensures that adjacent triangles on X are in correspondence with adjacent triangles on Y and therefore guarantees a discrete notion of continuity (see Figure 4.2). The natural discrete version of (i) is a closed, connected discrete surface in $X \times Y$.

Discrete version of (ii): As in the continuous setting, we can project product triangles to the surfaces X and Y by defining $\pi_X : F \to \mathbb{Z}^{|F_X|}$ as

$$(4.8) \qquad \pi_X(f) := \begin{cases} e_a & \text{if } a = \left(\begin{smallmatrix} a_1 \\ a_2 \\ a_3 \end{smallmatrix}\right) \text{ is non-deg.} \\ (0, \ldots, 0) & \text{else} \end{cases}$$

for each face $f = ((a_1, b_1), (a_2, b_2), (a_3, b_3)) \in F$. Here, e_a is the vector with 1 in the a-entry and 0 in all other entries. We extend the projection linearly to $\pi_X : \mathrm{surf}(X \times Y) \to \mathbb{Z}^{|F_X|}$.

Let now Γ be a discrete surface in $X \times Y$. Then we say that the projections of Γ to X and Y are discrete diffeomorphisms if and only if $\pi_X(\Gamma) = (1, \ldots, 1) \in \mathbb{Z}^{|F_X|}$ and $\pi_Y(\Gamma) = (1, \ldots, 1) \in \mathbb{Z}^{|F_Y|}$. This gives a discrete version of (ii).

Note that in this definition we do not ask for injectivity on the vertices set. This is necessary for modelling discretely strong compressions. However, they ensure a global bijectivity property which is sufficient in our context.

Discrete version of (iii): By definition, the set of surfaces in $X \times Y$ only contains surface patches which are consistently oriented. Therefore any surface in $\mathrm{surf}(X \times Y)$ satisfies condition (iii).

4.2.3. *Discrete surface energy.* Now we introduce a discrete energy on the set of product triangles in $X \times Y$. For the membrane energy in (2.6) we adopt the term proposed by Delingette [6]. Given two triangles $T_1, T_2 \subset \mathbb{R}^3$, Delingette computes the stretch energy $E_{\mathrm{mem}}(T_1 \to T_2)$ necessary for deforming T_1 in T_2. In our framework we make the energy symmetric by associating with each product triangle $(a, b) \in F$ the membrane cost $E_{\mathrm{mem}}(a, b) := E_{\mathrm{mem}}(a \to b) + E_{\mathrm{mem}}(b \to a)$. For the bending term we proceed similarly associating with each product triangle (a, b) the cost $E_{\mathrm{bend}}(a, b) = \int_a (H_X - H_Y)^2 + \int_b (H_Y - H_X)^2$. In practice we discretize the mean curvature following [19].

Next, we extend the energy linearly from discrete surface patches to discrete surfaces in $X \times Y$. Identify a discrete surface with its indicator vector $\Gamma \in \{0, 1\}^{|F|}$, and define the vector $E \in \mathbb{R}^{|F|}$ whose f-th entry is $E_f = E_{\mathrm{mem}}(f) + E_{\mathrm{bend}}(f)$. Then the *discrete energy* of Γ is given by the vector product $E^T \Gamma$.
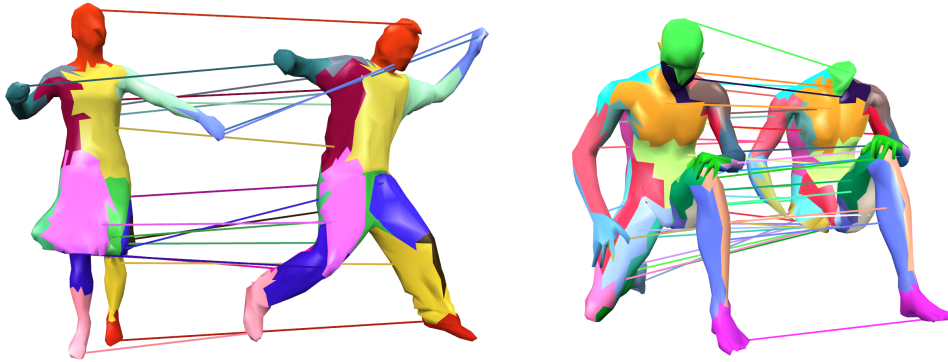
Figure 4.3: Matchings between 3D shapes obtained by minimizing the relaxation of energy (4.9). Since the energy can express elastic deformations such as stretching and shrinking the proposed method can find transformations that are highly non-rigid and non-isometric. The 3D shape data is from Vlasic et al. [29] (left) and the SHREC 2011 benchmark [2] (right).

4.2.4. *Optimizing the discrete energy.* The notion of discrete graph surfaces and the discrete surface energy introduced in Sections 4.2.2 and 4.2.3 can be combined with the discrete version of optimization problem (4.4) in the form of a binary linear program:

$$
(4.9) \quad
\begin{aligned}
&\min_{\Gamma \in \{0,1\}^{|F|}} \quad E^T \Gamma \\
&\text{subject to} \quad \begin{pmatrix} \partial \\ \pi_X \\ \pi_Y \end{pmatrix} \Gamma = \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \\ \mathbf{1} \end{pmatrix}.
\end{aligned}
$$

Similarly to what we did in (2.3), in order to solve (4.9) we relax the binary constraints to $\Gamma \in [0,1]^{|F|}$. For this relaxed version the global optimum can be computed in polynomial time. Since the constraint matrix of the relaxed problem is not totally unimodular, we are not guaranteed an integral solution. A simple thresholding scheme would destroy the geometric consistency of the solution. Therefore, for obtaining an integral solution we successively fix the variable with maximum value to 1. Typical matching results are shown in Figure 4.3. For a more detailed experimental evaluation we refer to [30].

5. Conclusions

In this paper we discussed two approaches to non-rigid shape matching by optimizing a distortion criterion. While for both approaches the distortion criterion is based on information derived from the metric of the shape, it is possible to express quite different notions of similarity, i.e. a geometrical Gromov-Wasserstein distance and a physical thin-shell energy. By following different algorithmic strategies for both notions of similarity, we showed that it is possible to find good matchings minimizing the distortion energies between the shapes.

Acknowledgments

References

[1] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge Univ. Press, New York, USA, 2004.

[2] E. Boyer, A. M. Bronstein, M. M. Bronstein, B. Bustos, T. Darom, R. Horaud, I. Hotz, Y. Keller, J. Keustermans, A. Kovnatsky, R. Litman, J. Reininghaus, I. Sipiran, D. Smeets, P. Suetens, D. Vandermeulen, A. Zaharescu, and V. Zobel. SHREC 2011: robust feature detection and description benchmark. *ArXiv e-prints*, February 2011.

[3] Alex Bronstein, Michael Bronstein, Umberto Castellani, et al. Shrec 2010: Robust correspondence benchmark. In *Eurographics Workshop on 3D Object Retrieval*, 2010.

[4] Alex Bronstein, Michael Bronstein, and Ron Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. National Academy of Science (PNAS)*, 103(5):1168–1172, 2006.

[5] P. Ciarlet. *An introduction to differential geometry with applications to elasticity*. Springer, Dordrecht, 2005. Reprinted from J. Elasticity **7**8/79 (2005), no. 1-3 [MR2196098].

[6] H. Delingette. Triangular springs for modeling nonlinear membranes. *IEEE Transactions on Visualisation and Computer Graphics*, 14(2), March/April 2008.

[7] M. Desbrun, A. N. Hirani, M. Leok, and J. E. Marsden. Discrete exterior calculus, 2005.

[8] M.P. Do Carmo. *Riemannian geometry*. Birkhauser, 1992.

[9] M. Giaquinta, G. Modica, and J. Souček. *Cartesian currents in the calculus of variations. II*, volume 38 of *Ergebnisse der Mathematik und ihrer Grenzgebiete.* Springer-Verlag, Berlin, 1998. Variational integrals.

[10] Leo Grady. Minimal surfaces extend shortest path segmentation methods to 3d. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(2):321–334, 2010.

[11] W. Koiter. On the nonlinear theory of thin elastic shells. I, II, III. *Nederl. Akad. Wetensch. Proc. Ser. B*, 69:1–17, 18–32, 33–54, 1966.

[12] Marius Leordeanu and Martial Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Proc. CVPR*, volume 2, pages 1482–1489, 2005.

[13] Y. Lipman and I. Daubechies. Surface comparison with mass transportation, 2009.

[14] Yaron Lipman and Thomas Funkhouser. Mobius voting for surface correspondence. *ACM Trans. on Graphics*, 28(3), 2009.

[15] N. Litke, M. Droske, M. Rumpf, and P. Schröder. An image processing approach to surface matching. In *Symposium on Geometry Processing*, pages 207–216, 2005.

[16] Eliane Maria Loiola, Nair Maria Maia de Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. A survey for the quadratic assignment problem. *European Journal of Operational Research*, 176(2):657–690, 2007.

[17] F. Mémoli and G. Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Found. Comput. Math.*, 5:313–346, 2005.

[18] Facundo Mémoli. Gromov-Wasserstein distances and the metric approach to object matching. *Found. Comput. Math.*, 11:417–487, 2011.

[19] M. Meyer, M. Desbrun, P Schröder, and A. Barr. Discrete differential-geometry operators for triangulated 2-manifolds, 2002.

[20] Maks Ovsjanikov, Qi-Xing Huang, and Leonidas J. Guibas. A condition number for non-rigid shape matching. *Comput. Graph. Forum*, pages 1503–1512, 2011.

[21] Emanuele Rodolà, Alex Bronstein, Andrea Albarelli, Filippo Bergamasco, and Andrea Torsello. A game-theoretic approach to deformable shape matching. In *Proc. CVPR*, 2012.

[22] Emanuele Rodolà, Tatsuya Harada, Yasuo Kuniyoshi, and Daniel Cremers. Efficient shape matching using vector extrapolation. In *Proc. BMVC*, 2013.

[23] Emanuele Rodolà, Andrea Torsello, Tatsuya Harada, Yasuo Kuniyoshi, and Daniel Cremers. Elastic net constraints for shape matching. In *Proc. ICCV*, Sydney, Australia, December 2013.

[24] F. R. Schmidt, Dirk Farin, and D. Cremers. Fast matching of planar shapes in sub-cubic runtime. In *IEEE Int. Conf. on Computer Vision*, Rio de Janeiro, October 2007.

[25] T. Schoenemann, F. Kahl, and D. Cremers. Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *IEEE Int. Conf. on Computer Vision*, Kyoto, 2009.

[26] Shai Shalev-Shwartz and Yoram Singer. Efficient learning of label ranking by soft projections onto polyhedra. *J. Mach. Learn. Res.*, 7:1567–1599, December 2006.

[27] John M. Sullivan. *A Crystalline Approximation Theorem for Hypersurfaces*. PhD thesis, Princeton University, October 1990.

[28] Hermant Tagare. Shape-based nonrigid correspondence with application to heart motion analysis. *IEEE Trans Med Imaging*, 18(7):570–579, 1999.

[29] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. Articulated mesh animation from multi-view silhouettes. In *ACM SIGGRAPH 2008 papers*, SIGGRAPH '08, pages 97:1–97:9, New York, NY, USA, 2008. ACM.

[30] Thomas Windheuser, Ulrich Schlickewei, Frank R Schmidt, and Daniel Cremers. Geometrically consistent elastic matching of 3d shapes: A linear programming solution. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2134–2141. IEEE, 2011.

[31] B. Wirth, L. Bar, M. Rumpf, and G. Sapiro. Geodesics in shape space via variational time discretization. In *EMMCVPR'09*, volume 5681 of *LNCS*, pages 288–302, 2009.

[32] Yun Zeng, Chaohui Wang, Yang Wang, Xianfeng Gu, Dimitris Samaras, and Nikos Paragios. Dense non-rigid surface registration using high-order graph matching. In *Proc. CVPR*, pages 382–389, 2010.

[33] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.

Technische Universität München • cremers@in.tum.de • rodola@in.tum.de • windheus@in.tum.de

Metric Ricci Curvature and Flow for *PL* Manifolds

Emil Saucan

Abstract

We summarize here the main ideas and results of our papers [28], [14], as presented at the 2013 CIRM Meeting on Discrete curvature and we augment these by bringing up an application of one of our main results, namely to solving a problem regarding cube complexes.

1. Introduction

While curvature itself represents a classical notion going back at least to Newton, it was traditionally restricted to the firmly established context of smooth (at least twice differentiable) curves and surfaces. It was however only natural that in the great furore of modernizing Mathematics, that enveloped the mathematical community in the first half of the XXth Century would include soon a drive of extending the notion of curvature to (quite) general metric spaces. This tendency came to fruit mainly in the works of Menger [22], Wald [35], [36] and Haantjes [16], [17]. While quickly falling into the desuetude (with very few, but notable exceptions, e.g. [20]), even if efforts [6], [7] were made in correctly re-ascertaining their importance and maintaining their influence, so to say, in other fields, a revival stated only towards the end of the previous century and metric curvatures began to reassume their rightful place in Geometry. This is particularly true as far as Menger's curvature is concerned – it has by now become a rather standard and successful tool in Analysis [24], [21], [31] (but not only - see, for instance [12]). In contrast, Haantjes curvature has received in recent literature far lesser attention, with a very few – but notable – exceptions, e.g. [1], where actualized, more elaborate versions of both Menger and Haantjes curvature are discussed in a contemporary framework. On the other hand, Haantjes curvature seems singularly well adapted for a multitude of applicative tasks – see [29], [30], [2], [28] (and perhaps even to applications in such fields as Geometric Group Theory). However, the best candidate for a successful incorporation of the metric metric approach to curvature into the main corpus of contemporary Mathematics, from Differential Geometry to Group Theory, is Wald curvature. Indeed, it turns out that Wald and Alexandrov (comparison) notion of curvature are essentially equivalent (at least for wide range of even mildly-well behaved spaces – see [25] (and also [6]). We tried to emphasize this equivalence and exploit it two our own specific goals in both papers we are summarizing here (and we shall further develop this theme in the the forthcoming lecture notes meant to accompany these proceedings and fully summarize the Colloquium.

2. Wald Metric Curvature – A Brief Overview

Wald's approach to the definition of a viable definition of curvature on (quite general) metric spaces was to mimic Gauss' original definition of (total) curvature. However, instead of making appeal to the comparison of infinitesimal areas (which would be unattainable in general metric space not endowed with a measure), he compared quadrangles. Moreover, instead in restricting himself solely to one comparison surface (namely the unit sphere $\mathbb{S}^n$), he considered the whole gamut of possible gauge surfaces, namely the surfaces S_κ, where S_κ denotes the complete, simply

connected surface of constant Gauss curvature κ, i.e. $S_\kappa \equiv \mathbb{R}^2$, if $\kappa = 0$; $S_\kappa \equiv \mathbb{S}^2_{\sqrt{\kappa}}$, if $\kappa > 0$; and $S_\kappa \equiv \mathbb{H}^2_{\sqrt{-\kappa}}$, if $\kappa < 0$. Here $S_\kappa \equiv \mathbb{S}^2_{\sqrt{\kappa}}$ denotes the sphere of radius $R = 1/\sqrt{\kappa}$, and $S_\kappa \equiv \mathbb{H}^2_{\sqrt{-\kappa}}$ stands for the hyperbolic plane of curvature $\sqrt{-\kappa}$, as represented by the Poincaré model of the plane disk of radius $R = 1/\sqrt{-\kappa}$. We will make this clear in the following sequence of definitions:

Definition 1. Let (M, d) be a metric space, and let $Q = \{p_1, ..., p_4\} \subset M$, together with the mutual distances: $d_{ij} = d_{ji} = d(p_i, p_j)$; $1 \le i, j \le 4$. The set Q together with the set of distances $\{d_{ij}\}_{1 \le i,j \le 4}$ is called a *metric quadruple*.

Definition 2. The *embedding curvature* $\kappa(Q)$ of the metric quadruple Q is defined to be the curvature κ of the gauge surface S_κ into which Q can be isometrically embedded. (See Figure 1.)
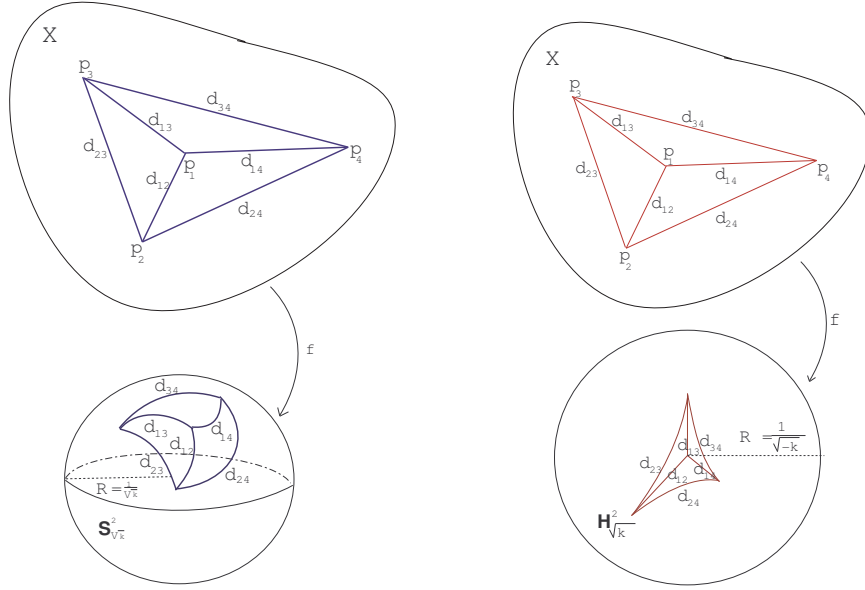
Figure 2.1: Isometric embedding of a metric quadruple in a gauge surface: $\mathbb{S}^2_{\sqrt{\kappa}}$ (left) and $\mathbb{H}^2_{\sqrt{\kappa}}$ (right).

We are now able to bring the definition of *Wald curvature* [35],[36] (or, more precisely, a slight modification of it due to Berestovskii [4]):

Definition 3. Let (X, d) be a metric space. An open set $U \subset X$ is called a *region of curvature* $\ge \kappa$ iff any metric quadruple can be isometrically embedded in S_m, for some $m \ge \kappa$.[1]

Remark 4. Evidently, in the context of polyhedral surfaces, the natural choice for the set U required in Definition 3 is the *star* of a given vertex v, that is, the set $\{e_{v_j}\}_j$ of edges incident to v. Therefore, for such surfaces, the set of metric quadruples containing the vertex v is finite.

Equipped with this quite simple and intuitive choice for U (and in in analogy with Alexandrov spaces – see also [14]) it is quite natural to consider, for *PL* surfaces, the following definition of the Wald curvature $K(v)$ at the vertex v:

$$K_W(v) = \min_{v_i, v_j, v_k \in \mathrm{Lk}(v)} K_W^{ijk}(v),$$

where $K_W^{ijk}(v) = \kappa(v; v_i, v_j, v_k)$, and where $\mathrm{Lk}(v)$ denotes the *link* of the vertex v – see Footnote 2 below.

Here we consider the (intrinsic) *PL* distance between vertices.

---

[1] While this fact is not needed in the remainder of the paper, we mention for the sake of completeness, that a metric space (X, d) is said to have *Wald-Berestovskii curvature* $\ge \kappa$ iff any $x \in X$ is contained in a region U of curvature $\ge \kappa$.

The definition of the Wald-Berestovskii curvature at an accumulation point of a metric space, hence on a smooth surface, follows naturally by considering limits of the curvatures of (nondegenerate) regions of diameter converging to 0. Moreover, the following important and "reassuring" result holds:

Theorem 5 (Wald [36]). *Let $S \subset \mathbb{R}^3$, $S \in \mathcal{C}^m$, $m \geq 2$ be a smooth surface. Then, given $p \in S$, $\kappa_W(p)$ exists and $\kappa_W(p) = K(p)$, where $K(p)$ denotes the Gaussian curvature at p.*

A further fact that makes Wald curvature attractive for Discrete Differential Geometry applications is the existence of a concrete (albeit somewhat impractical) computation formula:

Given a metric quadruple Q,

$$(2.1) \qquad \kappa(Q) = \begin{cases} 0 & \text{if } \Gamma(Q) = 0 \,; \\ \kappa, \, \kappa < 0 & \text{if } det(\cosh \sqrt{-\kappa} \cdot d_{ij}) = 0 \,; \\ \kappa, \, \kappa > 0 & \text{if } det(\cos \sqrt{\kappa} \cdot d_{ij}) \text{ and } \sqrt{\kappa} \cdot d_{ij} \leq \pi \\ & \qquad \text{and all the principal minors of order 3 are } \geq 0; \end{cases}$$

where $d_{ij} = d(p_i, p_j), 1 \leq i, j \leq 4$, and $\Gamma(Q) = \Gamma(p_1, \ldots, p_4)$ denotes the Cayley-Menger determinant:

$$(2.2) \qquad \Gamma(p_0, \ldots, p_3) = \begin{vmatrix} 0 & d_{01}^2 & \cdots & d_{13}^2 \\ d_{10}^2 & 0 & \cdots & d_{13}^2 \\ \vdots & \vdots & \ddots & \vdots \\ d_{30}^2 & d_{31}^2 & \cdots & 0 \end{vmatrix} .$$

3. Metric Ricci Flow for *PL* Surfaces

Our approach to this problem is to pass from the discrete context to the smooth one and explore the already classical results known in this setting, by applying Theorem 5. To this end we have first to make a few observations: One can pass from the *PL* surfaces to smooth ones by employing *smoothings*, defined in the precise sense of *PL* differential Topology (see [23]). Since, by [23], Theorem 4.8, such smoothings are *δ-approximations*, and therefore, for δ small enough, also α-approximations of the given piecewise-linear surface S_{Pol}^2, they approximate arbitrarily well both distances and angles on S_{Pol}^2. (Due to space restrictions, we do not bring here these technical definitions, but rather refer the reader to [23].) It should be noted that, while Munkres' results concern *PL* manifolds, they can be applied to polyhedral ones as well, because, by definition, polyhedral manifolds have simplicial subdivisions (and furthermore, such that all vertex links[2] are combinatorial manifolds). Of course, for different subdivisions, one may obtain different polyhedral metrics. However, by the *Hauptvermutung* Theorem in dimension 2 (and, indeed, for smooth triangulations of diffeomorphic manifolds in any dimension, see e.g. [23] and the references therein), any two subdivisions of the same space will be combinatorially equivalent, hence they will give rise to the same polyhedral metric. It follows from the observations above that *metric quadruples* on S_{Pol} are also arbitrarily well approximated (including their angles) by the corresponding metric quadruples) on S_m. But, by Theorem 5, $K_{W,m}(p)$ – the *Wald metric curvature* of S_m, at a point p – equals the classical (Gauss) curvature $K(p)$. Hence the Gauss curvature of the smooth surfaces S_m approximates arbitrarily well the metric one of S_{PL} (and, as in [8], the smooth surfaces differ from polyhedral one only on (say) the $\frac{1}{m}$-neighbourhood of the 1-skeleton of S_{Pol}). Moreover, this statement can be made even more precise, by assuring that the convergence is in the Hausdorff metric. This follows from results of Gromov (see e.g. [30] for details).

We can now introduce the *metric* Ricci flow: By analogy with the classical flow

$$(3.1) \qquad \frac{dg_{ij}(t)}{dt} = -2K(t)g_{ij}(t) \,.$$

---

[2]Recall that the *link* $\mathrm{lk}(v)$ of a vertex v is the set of all the faces of $\overline{\mathrm{St}}(v)$ that are not incident to v. Here $\overline{\mathrm{St}}(v)$ denotes the *closed star* of v, i.e. the smallest subcomplex (of the given simplicial complex K) that contains $\mathrm{St}(v)$, namely $\overline{\mathrm{St}}(v) = \{\sigma \in \mathrm{St}(v)\} \cup \{\theta \,|\, \theta \leqslant \sigma\}$, where $\mathrm{St}(v)$ denotes the *star* of v, that is the set of all simplices that have v as a face, i.e $\mathrm{St}(v) = \{\sigma \in K \,|\, v \leqslant \sigma\}$.

we define the *metric* Ricci flow by

$$\frac{dl_{ij}}{dt} = -2K_i l_{ij} \,, \tag{3.2}$$

where $l_{ij} = l_{ij}(t)$ denote the edges (1-simplices) of the triangulation (*PL* or piecewise flat surface) incident to the vertex $v_i = v_i(t)$, and $K_i = K_i(t)$ denotes the Wald curvature at the same vertex.

Moreover, we also consider the close relative of (3.1), the normalized flow

$$\frac{dg_{ij}(t)}{dt} = (K - K(t))g_{ij}(t) \,, \tag{3.3}$$

and its metric counterpart

$$\frac{dl_{ij}}{dt} = (\bar{K} - K_i)l_{ij} \,, \tag{3.4}$$

where $K, \bar{K}$ denote the average classical, respectively Wald, sectional (Gauss) curvature of the initial surface S_0: $K = \int_{S_0} K(t)dA / \int_{S_0} dA$, and $\bar{K} = \frac{1}{|V|} \sum_{i=1}^{|V|} K_i$, respectively. (Here $|V|$ denotes, as usually, the cardinality of the vertex set of S_{Pol}.)

Before continuing further on, it is important to remark the asymmetry in equation 3.2, that is caused by the fact that the curvature on two different vertices acts, so to say, on the same edge. However, passing to the smooth case, is that the asymmetry in the metric flow that we observed above disappears automatically via the smoothing process. (For further details see [28].)

3.1. **An Approximation Result.** The first result that we can bring is a metric curvature version of classical result of Brehm and Kühnel [8] (where the combinatorial/defect definition of curvature for polyhedral surfaces is used).

Proposition 6. *Let S_{Pol}^2 be a compact polyhedral surface without boundary. Then there exists a sequence $\{S_m^2\}_{m\in\mathbb{N}}$ of smooth surfaces, (homeomorphic to S_{Pol}^2), such that*
  (1) (a) *$S_m^2 = S_{Pol}^2$ outside the $\frac{1}{m}$-neighbourhood of the 1-skeleton of S_{Pol}^2,*
      (b) *The sequence $\{S_m^2\}_{m\in\mathbb{N}}$ converges to S_{Pol}^2 in the Hausdorff metric;*
  (2) *$K(S_m^2) \to K_W(S_{Pol}^2)$, where the convergence is in the weak sense.*

Remark 7. As we have already noted above, the converse implication – namely that Gaussian curvature $K(\Sigma)$ of a smooth surface Σ may be approximated arbitrarily well by the Wald curvatures $K_W(\Sigma_{Pol,m})$ of a sequence of approximating polyhedral surfaces $\Sigma_{Pol,m}$ – is quite classical.

For a more in-depth discussion and analysis of the convergence rate in the proposition above, see [28].

3.2. **Main Results.** As already stressed, the "good", i.e. metric and curvature, approximations results quoted, imply that one can study the properties of the metric Ricci flow via those of its classical counterpart, by passing to a smoothing of the polyhedral surface. The use of the machinery of metric curvature considered has the benefit that, by using it, the "duality" between the combinatorics of the packings (and angles) and the metric disappears: The flow becomes purely metric and, moreover, the curvature at each stage (i.e. for every "t") is given – as in the smooth setting – in an intrinsic manner, that is in terms of the metric alone.

We bring here a number of important properties that follow immediately using this approach.

3.2.1. *Existence and Uniqueness.* The main result that we can state here (and, in fact, in this section) is

Proposition 8. *Let (S_{Pol}^2, g_{Pol}) be a compact polyhedral 2-manifold without boundary, having bounded metric curvature. Then there exists $T > 0$ and a smooth family of polyhedral metrics $g(t), t \in [0, T]$, such that*

$$\begin{cases} \frac{\partial g}{\partial t} = -2K_W(t)g(t) & t \in [0, T] \,; \\ g(0) = g_{Pol} \,. \end{cases} \tag{3.5}$$

(Here $K_W(t)$ denotes the Wald curvature induced by the metric $g(t)$.)

Moreover, both the forwards and the backwards (when existing) Ricci flows have the uniqueness of solutions property, that is, if $g_1(t), g_2(t)$ are two Ricci flows on S_{Pol}^2, such that there exists $t_0 \in [0, T]$ such that $g_1(t_0) = g_2(t_0)$, then $g_1(t) = g_2(t)$, for all $t \in [0, T]$.

Beyond the theoretical importance, the existence and uniqueness of the backward flow would allow us to find surfaces in the conformal class of a given circle packing (Euclidean or Hyperbolic). More importantly, the use of purely metric, Wald curvature based, approach adopted, rather than the combinatorial (and metric) approach of [11], allows us to give a preliminary and purely theoretical at this point, answer to Question 2, p. 123, of [11], namely whether there exists a Ricci flow defined on the space of all piecewise constant curvature metrics (obtained via the assignment of lengths to a given triangulation of 2-manifold). Since, by the results of Hamilton [16] and Chow [10], the Ricci flow exists for all compact surfaces, it follows that the fitting metric flow exists for surfaces of piecewise constant curvature. In consequence, given a surface of piecewise constant curvature (e.g. a mesh with edge lengths satisfying the triangle inequality for each triangle), one can evolve it by the Ricci flow, either forward, as in the works discussed above, to obtain, after the suitable area normalization, the polyhedral surface of constant curvature conformally equivalent to it; or backwards (if possible) to find the "primitive" family of surfaces – including the "original" surface obtained via the backwards Ricci flow, at time T – conformally equivalent to the given one.

3.2.2. *Convergence Rate.* A further type of result, quite important both from the theoretical viewpoint and for computer-driven applications, is that of the *convergence rate* (see [15], [14] for the precise definition).

Since we already know that the solution exists and it is unique (see also the subsection below for the nonformation of singularities), by appealing to the classical results of [16] and [10], we can control the convergence rate of the curvature, as follows:

Theorem 9. *Let* (S^2_{Pol}, g_{Pol}) *be a compact polyhedral 2-manifold without boundary. Then the normalized metric Ricci flow converges to a surface of constant metric curvature. Moreover, the convergence rate is*

(1) *exponential, if* $\bar{K} = \bar{K}_W < 0$ *(i.e.* $\chi(S^2_{Pol}) < 0$*) ;*
(2) *uniform; if* $\bar{K} = 0$ *(i.e.* $\chi(S^2_{Pol}) = 0$*);*
(3) *exponential, if* $\bar{K} > 0$ *(i.e.* $\chi(S^2_{Pol}) > 0$*).*

3.2.3. *Singularities Formation.* Another very important aspect of the Ricci flow, both smooth or discrete, is that of singularities formation. Again, a certain (theoretical, at least) advantage of the proposed method presents itself. Indeed, by [11], Theorem 5.1, for compact surfaces of genus ≥ 2, the combinatorial Ricci flow evolves without singularities. However, for surfaces of low genus no such result exists. Indeed, in the case of the Euclidean background metric, that is the one of greatest interest in graphics, singularities do appear. Moreover, such singularities are always combinatorial in nature and amount to the fact that, at some t, the edges of at least one triangle do not satisfy the triangle inequality. These singularities are removed in heuristic manner. However, by [16], Theorem 1.1, the smooth Ricci flow exists at all times, i.e. no singularities form. From the considerations above, it follows that the metric Ricci flow also exists at all times without the formation of singularities.

3.2.4. *Embeddability in* $\mathbb{R}^3$. The importance of the embeddability of the flow is not merely theoretical (e.g. if one considers the problem of the Ricci flow for surfaces of piecewise constant curvature), as it is essential in Imaging (see [3]), and of high importance in Graphics. Indeed, even our very capability of seeing (grayscale) images is nothing but a translation, in the field of vision, of the embeddability of the associated height-surface into $\mathbb{R}^3$. (Or, perhaps one should view the mathematical aspect as a formalization of a physical/biological phenomenon...) We should note here that in this respect there exists a certain (mainly theoretical, at this point in time) advantage of our proposed metric flow over the combinatorial Ricci flow [15], [19]. Indeed, in the combinatorial flow, the goal is to produce, via the circle packing metric, a conformal mapping from the given surface to a surface of constant (Gauss) curvature. Since in the relevant cases (see [11]) the surface in question is a planar region (usually a subset of the unit disk), its embeddability (not necessarily isometric) is trivial. Moreover, in the above mentioned works, there is no interest (and indeed, no need) to consider the (isometric) embeddability of the surfaces S^2_t (see below) for an intermediate time $t \neq 0, T$.

The tool that allows us to obtain this type of results is making appeal (again) to δ-approximations, in combination with classical results in embedding theory. Indeed, by [23], Theorem 8.8 a δ-approximation of an embedding is also an embedding, for small enough δ. Since, as we have already mentioned, smoothing represent δ-approximations, the possibility of using results regarding smooth surfaces to infer results regarding polyhedral embeddings is proven. (The other direction – namely from smooth to *PL* and polyhedral manifolds – follows from the fact that the *secant approximation* is a δ-approximation if the simplices of the *PL* approximation satisfy a certain nondegeneracy condition – see [23], Lemma 9.3.) We state here the relevant facts:

Let S_0^2 be a smooth surface of positive Gauss curvature, and let S_t^2 denote the surface obtained at time t from S_0^2 via the Ricci flow. (For all omitted background material – proofs, further results, etc. – we refer to [18].)

Proposition 10. *Let S_0^2 be the unit sphere $\mathbb{S}^2$, equipped with a smooth metric g, such that $\chi(S_0^2) > 0$. Then the surfaces S_t^2 are (uniquely, up to a congruence) isometrically embeddable in $\mathbb{R}^3$, for any $t \geq 0$.*

In fact, this results can be slightly strengthened as follows:

Corollary 11. *Let S_0^2 be a compact smooth surface. If $\chi(S_0^2) > 0$, then there exists some $t_0 \geq 0$, such that the surfaces S_t^2 are isometrically embeddable in $\mathbb{R}^3$, for any $t \geq t_0$.*

In stark contrast with this positive result regarding surfaces uniformized by the sphere, for (complete) surfaces uniformized by the hyperbolic plane we only have the following negative result:

Proposition 12. *Let (S_0^2, g_0) be a complete smooth surface, and consider the normalized Ricci flow on it. If $\chi(S_0^2) < 0$, then there exists some $t_0 \geq 0$, such that the surfaces S_t^2 are not isometrically embeddable in $\mathbb{R}^3$, for any $t \geq t_0$.*

4. Metric Ricci Curvature for *PL* Manifolds

We propose a definition of a metric Ricci curvature for *PL* manifolds in dimension higher than 2, that does not appeals to smoothings, as in the previous section.

4.1. **The Definition.** While the results in the preceding sections might be encouraging, one would still like to recover in the metric setting a "full" Ricci curvature, namely one that holds for 3 and higher dimensional manifolds, and not just in the degenerate case of surfaces. Our approach (as developed in [14]) is to start from the following classical formula:

$$(4.1) \qquad \mathrm{Ric}(e_1) = \mathrm{Ric}(e_1, e_1) = \sum_{i=2}^{n} K(e_1, e_i) \, .$$

for any orthonormal basis $\{e_1, \cdots, e_n\}$, and where $K(e_1, e_j)$ denotes the sectional curvature of the 2-sections containing the directions e_1.

To adapt this expression for the Ricci curvature to the *PL* case, we first have to be able to define (*variational*) *Jacobi fields*. In this we heavily rely upon Stones's work [32], [33]. Note, however, that we do not need the full strength of Stone's technical apparatus, only the capability determine the relevant two sections and, of course, to decide what a direction at a vertex of a *PL* manifold is.

We start from noting that, in Stone's work, combinatorial Ricci curvature is defined both for the given simplicial complex $\mathcal{T}$, and for its *dual complex* $\mathcal{T}^*$. For the dual complex, cells – playing here the role of the planes in the classical setting of which sectional curvatures are to be averaged – are considered. Unfortunately, Stone's approach for the given complex, where one computes the Ricci curvature $\mathrm{Ric}(\sigma, \tau_1 - \tau_2)$ of an n-simplex σ in the direction of two adjacent $(n-1)$-faces, τ_1, τ_2, is not natural in a geometric context (even if useful in his purely combinatorial one), except for the 2-dimensional case, where it coincides with the notion of Ricci curvature in a direction. However, passing to the dual complex will not restrict us, since $(\mathcal{T}^*)^* = \mathcal{T}$ and, moreover – and more importantly – considering *thick* triangulations enables us to compute the more natural metric curvature for the dual complex and use the fact that the dual of a thick triangulation is thick (for details, see [14]). Recall that thick (also called *fat*) triangulations are defined as follows:

Definition 13. Let $\tau \subset \mathbb{R}^n$; $0 \le k \le n$ be a k-dimensional simplex. The *thickness* (or *fatness*) φ of τ is defined as being:

$$(4.2) \qquad \varphi(\tau) = \frac{\operatorname{dist}(b, \partial\sigma)}{\operatorname{diam}\sigma},$$

where b denotes the *barycenter* of σ and $\partial\sigma$ represents the standard notation for the *boundary* of σ (i.e the union of the $(n-1)$-dimensional faces of σ). A simplex τ is φ_0-*thick*, for some $\varphi_0 > 0$, if $\varphi(\tau) \ge \varphi_0$. A triangulation (of a submanifold of $\mathbb{R}^n$) $\mathcal{T} = \{\sigma_i\}_{i\in\mathbf{I}}$ is φ_0-*thick* if all its simplices are φ_0-thick. A triangulation $\mathcal{T} = \{\sigma_i\}_{i\in\mathbf{I}}$ is *thick* if there exists $\varphi_0 \ge 0$ such that all its simplices are φ_0-*thick.*

Keeping in mind the notions and facts above, we can now return to the definition of Ricci curvature for simplicial complexes: Given a vertex v_0 in the dual complex, corresponding to a n-dimensional simplicial complex, a *direction* at v_0 is just an oriented edge $e_1 = v_0 v_1$. Since there exist precisely n 2-cells, $\mathfrak{c}_1, \ldots, \mathfrak{c}_n$, having e_1 as an edge and, moreover, these cells form part of n relevant variational (Jacobi) fields (see [32]), the Ricci curvature at the vertex v, in the direction e_1 is simply

$$(4.3) \qquad \operatorname{Ric}(v) = \sum_{i=1}^{n} K(\mathfrak{c}_i),$$

where we define the sectional curvature of a cell $\mathfrak{c}$ in the following manner:

Definition 14. Let $\mathfrak{c}$ be a cell with vertex set $V_{\mathfrak{c}} = \{v_1, \ldots, v_p\}$. The *embedding curvature* $K(\mathfrak{c})$ of $\mathfrak{c}$ is defined as:

$$(4.4) \qquad K(\mathfrak{c}) = \min_{\{i,j,k,l\} \subseteq \{1,\ldots,p\}} \kappa(v_i, v_j, v_k, v_l).$$

Remark 15. Note that by choosing to work with the dual complex we have restricted ourselves largely to considering solely submanifolds of $\mathbb{R}^N$, for some N sufficiently large. However, in the case of 2-dimensional *PL* manifolds this does nor represent restriction, since, by a result of Burago and Zalgaller [9] (see also [27]) such manifolds admit isometric embeddings in $\mathbb{R}^3$, embeddings that, furthermore, are unique (up to isometries of the ambient space, of course).

Remark 16. Evidently, the definition above presumes that cells in the dual complex have at least 4 vertices. However, except for some utterly degenerate (planar) cases, this condition always holds. Still, even in this case Ricci curvature can be computed using a slightly different approach – see the following remark.

Remark 17. It is still possible (by dualization) to compute Ricci curvature according, more or less, to Stone's ideas, at least for the 2-dimensional case. Indeed, according to [33],

$$(4.5) \qquad \operatorname{Ric}(\sigma, \tau_1 - \tau_2) = 8n - \sum_{j=1}^{2n-1} \{|\beta_j| \mid \beta_j < \tau_1 \text{ or } \beta_j < \tau_2; \dim \beta_j = n - 2\}.$$

For details and implications of this alternative approach, see [14].

4.2. **Main Results.** The first results one wants to ascertain are those ensuring the convergence of the newly defined Ricci curvature. These are quite straightforward, so here we content ourselves with simply stating them (for further details, see [14]).

Theorem 18. *Let $\mathcal{T}$ be a thick simplicial complex, and let $\mathcal{T}^*$ denote his dual. Then*

$$(4.6) \qquad \lim_{\operatorname{mesh}(\mathcal{T})\to 0} \operatorname{Ric}(\sigma) = \lim_{\operatorname{mesh}(\mathcal{T}^*)\to 0} C \cdot \operatorname{Ric}^*(\sigma^*),$$

where $\sigma \in \mathcal{T}$ and where $\sigma^* \in \mathcal{T}^*$ is (as suggested by the notation) the dual of σ.

Theorem 19. *Let M^n be a (smooth) Riemannian manifold and let $\mathcal{T}$ be a thick triangulation of M^n. Then*

$$(4.7) \qquad \operatorname{Ric}_{\mathcal{T}} \to C_1 \cdot \operatorname{Ric}_{M^n}, \text{ as } \operatorname{mesh}(\mathcal{T}) \to 0,$$

where the convergence is the weak convergence (of measures).

Beyond these convergence and approximations results, one would like to address deeper issues. Indeed, having introduced a metric Ricci curvature for *PL* manifolds, one naturally wishes to verify that this represents a proper notion of Ricci curvature, and not just an approximation of the classical notion. According to the synthetic approach to Differential Geometry, a proper notion of Ricci curvature should satisfy adapted versions of the main, essential theorems that hold for the classical notions. The first and foremost among such theorems is Bonnet-Myers' Theorem and, as expected, fitting versions for combinatorial cell complexes and weighted cell complexes were proven by Stone [32], [33], [34].

In [14] we proved a series of increasingly more general variants of the Bonnet-Myers Theorem, with proofs adapted to the various settings and/or notions of curvature (metric, combinatorial, Alexandrov comparison). Here we bring only two more representative ones.

Theorem 20 (*PL* Bonnet-Myers – metric). *Let M_{PL}^n be a complete, n-dimensional* PL*, smoothable manifold without boundary, such that*

(i) There exists $d_0 > 0$, such that $\mathrm{mesh}(M_{PL}^n) \leq d_0$;

(ii) $K_W(M_{PL}^n) \geq K_0 > 0$,

where $K_W(M_{PL}^n)$ denotes the sectional curvature of the "combinatorial 2-sections".

Then M_{PL}^n is compact and, moreover

$$(4.8) \qquad \mathrm{diam}(M_{PL}^n) \leq \frac{\pi}{\sqrt{K_0}}.$$

Unfortunately, determining whether a general *PL* complex has Wald curvature bounded from below can be, in practice, a daunting task. However, in the special case of thick complexes in $\mathbb{R}^N$, for some N one can determine a simple criterion as follows:

Theorem 21 (*PL* Bonnet-Myers – Thick Complexes). *Let $M = M_{PL}^n$ be a complete, connected* PL *manifold thickly embedded in some $\mathbb{R}^N$, such that $K_W(M^2) \geq K_0 > 0$, where M^2 denotes the 2-skeleton of M. Then M_{PL}^n is compact and, moreover*

$$(4.9) \qquad \mathrm{diam}(M_{PL}^2) \leq \frac{\pi}{\sqrt{K_0}}.$$

4.3. **Scalar Curvature and a Comparison Theorem.** Up to this point we were concerned solely with Ricci curvature (as the very title stresses). However, since Ricci curvature is the mean of sectional curvatures we had to consider them too (and, in fact, even more so in view of our definition of Ricci curvature for *PL* complexes). We did not discuss, however, scalar curvature. It is only fitting, therefore, for us to add a number of observation regarding this invariant, in particular since a significant result is very easy to formulate and prove.

Of course, we first have to define the scalar curvature $K_W(M)$ of a *PL* manifold M. In light of our preceding discussion and results, the following definition is quite natural:

Definition 22. Let $M = M_{PL}^n$ be an n-dimensional *PL* manifold (without boundary). The *scalar metric curvature* scal_W of M is defined as

$$(4.10) \qquad \mathrm{scal}_W(v) = \sum_{\mathfrak{c}} K_W(\mathfrak{c}),$$

the sum being taken over all the cells of M^* incident to the vertex v of M^*.

Remark 23. Observe that the definition of scalar curvature of M is defined, somewhat counterintuitively, by passing to its dual M^*. However, this is consistent with our approach to Ricci curvature (and also similar to Stone's original approach – see the discussion in 4.1 above).

From this definition and our previous results (see [14]), we immediately[3] obtain, the following generalization of the classical curvature bounds comparison in Riemannian geometry:

Theorem 24 (Comparison theorem). *Let $M = M_{PL}^n$ be an n-dimensional* PL *manifold (without boundary), such that $K_W(M) \geq K_0 > 0$, i.e. $K(\mathfrak{c}) \geq K_0$, for any 2-cell of the dual manifold (cell complex) M^*. Then*

$$(4.11) \qquad K_W \underset{>}{\overset{\leq}{=}} K_0 \Rightarrow \mathrm{Ric}_W \underset{>}{\overset{\leq}{=}} nK_0.$$

---

[3]and, in truth rather trivially, since the result holds, regardless of the specific definition for the curvature of a cell.

Moreover

(4.12) $$K_W \lesseqgtr K_0 \Rightarrow \text{scal}_W \lesseqgtr n(n+1)K_0\,.$$

Remark 25.

(1) Inequality (4.12) can be formulated in the seemingly weaker form:

(4.13) $$\text{Ric}_W \lesseqgtr nK_0 \Rightarrow \text{scal}_W \lesseqgtr n(n+1)K_0\,.$$

(2) Note that in all the inequalities above, the dimension n appears, rather then $n-1$ as in the smooth, Riemannian case (hence, for instance one has in (4.12), $n(n+1)K_0$, instead of $n(n-1)K_0$[4] as in the classical case). This is due to our definition (4.3) of Ricci (and scalar) curvature, via the dual complex of the given triangulation, hence imposing standard and simple combinatorics, at the price of allowing only for such weaker bounds.[5]

5. An Application: Smoothable Metrics on Cube Complexes

In this last section we illustrate our belief in the many possible uses of the metric Ricci flow with only (due to space and time restrictions) one example, appertaining to the corpus of "Pure" Mathematics. The following seemingly well known problem in the theory of cube complexes[6] was posed to the author by Joel Haas, together with the basic idea of the first part of the proof, for which the author is deeply grateful.

Let $\mathcal{C}$ be a cube complex, satisfying the following conditions:

(1) $\mathcal{C}$ is negatively curved (i.e. such that $\#_v Q \geq 4$, for all vertices v,where $\#_v Q$ denotes the number of cubes incident to the vertex v;
(2) The link $\text{lk}(v)$ of any vertex is a *flag complex*, i.e. a simplicial complex such that any 3-arcs closed curve bounds a triangle (2-simplex), i.e. no such curve separates without being a boundary.[7]

Question 26. Does there exist a Riemannian metric g (on $\mathcal{C}$) such that $K_g \equiv K$, where K denotes the comparison (Alexandrov) curvature of $\mathcal{C}$?

In other words: Does there exist a smoothing of (M,g) (i.e. Riemannian manifold) of a given cube complex $\mathcal{C}$ (that has a manifold structure), such that $K \equiv K_g$? Evidently, an important particular case would be that "cubical version of *PL* approximations"), i.e. that of "cubulations" of a (given) Riemannian manifold.

Remark 27. The similar problem can be also posed, of course, for positively curved complexes (i.e. such that $\#_v Q \leq 4$). However, we address here only the negatively curved case.

Clearly the answer to Question 26 above is "No", even if $\mathcal{C}$ is actually a manifold, since it is not always possible to recover the Riemannian metric from the discrete ("cubical") one. (Recall that each edge is supposed to be of length 1.) However, in the special case of 3-dimensional cube complexes the question has a positive answer.

We sketch below the proof:

(1) *Away from the vertices*, i.e. around the edges,[8] one can use a method developed by Gromov and Thurston [13] to produce a generalized type of branched cover (in any dimension). More precisely, (a) construct negatively curved conical surfaces of revolution, with vertex at a vertex v and with apex angle $\alpha = 2\pi/n$, where $n = \#_v Q$. Each such cone can be canonically mapped upon a Euclidean cone of apex angles $\pi/2$; then (b) glue the outcome of this process to the result of Step (2) below.

---

[4]but, on the other hand, this holds even if $n=3$!...

[5]without affecting the analogue of the Bonnet-Myers Theorem – see Section 2 above.

[6]For a formal definition and more details see, e.g. [26].

[7]Alternatively, this condition may be expressed either as $\text{lk}(v)$ "has no missing simplices (M. Sageev, [26]), or as "a nonsimplex contains a non edge" (W. Dicks, see [5]).

[8]obviously, in the interiors of the faces the metric is already smooth.

(2) *Around the vertices* excise an ε-ball neighbourhood B_ε of v. On the boundary of B_ε, i.e. on the sphere S_ε one has the natural triangulation by the intersections of S_ε with the cubes of $\mathcal{C}$ incident with v. Moreover, the curvature of the vertices of this triangulation will be $K_\varepsilon \equiv c/\varepsilon^2$, where c is some constant. However, while the gluing itself is trivial, one still has to ensure that the result is indeed endowed with a Riemannian metric. For this one has to go through Step 3 of the construction:

(3) *Smoothen the ball B_ε.* In general dimension this represents a daunting problem. Indeed, even in dimension 3, Ricci flow – who represents a natural candidate for smoothing with control of curvature – is yet not attainable, since we can offer, at this point in time, no *PL* (metric) Ricci flow. However, due to Perelman's resolution of the Poincaré conjecture, in dimension 3 suffices to smoothen the boundary S_ε. It is at this point where the method described in Section 3 is applied, producing the required smooth ball $\tilde{S}_\varepsilon$, that has the same curvature as the PL^9 one S_ε.

Acknowledgements

References

[1] S. B. Alexander and R. L. Bishop. Comparison theorems for curves of bounded geodesic curvature in metric spaces of curvature bounded above. *Differential Geometry and its Applications*, 6(10):67–86, 1996.

[2] E. Appleboim, Y. Hyams, S. Krakovski, C. Sageev, and E. Saucan. The scale-curvature connection and its application to texture segmentation. *Theory and Applications of Mathematics & Computer Science*, 3(1):38–54, 2013.

[3] E. Appleboim, E. Saucan, and Y. Y. Zeevi. Ricci curvature and flow for image denoising and superesolution. In *Proceedings of EUSIPCO 2012*, pages 2743–2747, 2012.

[4] V. N. Berestovskii. Spaces with bounded curvature and distance geometry. *Siberian Math. J.*, 16(1):8–19, 1986.

[5] M. Bestvina. Geometric group theory and 3-manifolds hand in hand: the fulfillment of thurston's vision. *Bull. Amer. Math. Soc.*, 51(1):53–70, 2014.

[6] L. M. Blumenthal. *Distance Geometry – Theory and Applications*. Claredon Press, Oxford, 1953.

[7] L. M. Blumenthal and K. Menger. *Studies in Geometry*. Freeman and Co., San Francisco, CA, 1970.

[8] U. Brehm and W. Kühnel. Smooth approximation of polyhedral surfaces regarding curvatures. *Geometriae Dedicata*, 12(1):435–461, 1982.

[9] Yu. D. Burago and V. A. Zalgaller. Isometric piecewise linear immersions of two-dimensional manifolds with polyhedral metrics into $\mathbb{R}^3$. *St. Petersburg Math. J.*, 7(3):369–385, 1996.

[10] B. Chow. The ricci flow on the 2-sphere. *J. Differential Geom.*, 33(2):325–334, 1991.

[11] B. Chow and F. Luo. Combinatorial ricci flows on surfaces. *J. Differential Geom.*, 63(1):97–129, 2003.

[12] J. Giesen. Curve reconstruction, the traveling salesman problem and menger's theorem on length. In *Proceedings of the 15th ACM Symposium on Computational Geometry (SoCG)*, pages 207–216, 1999.

[13] M. Gromov and W. Thurston. Pinching constants for hyperbolic manifolds. *Invent. math.*, 89(10):1–12, 1987.

[14] D. X. Gu and E. Saucan. Metric ricci curvature for *pl* manifolds. *Geometry*, page 12 pages, 2013.

[15] X. D. Gu and S.-T. Yau. *Computational Conformal Geometry*. Advanced Lectures in Mathematics **3**, International Press, Somerville, MA, 2008.

[16] J. Haantjes. Distance geometry. curvature in abstract metric spaces. *Proc. Kon. Ned. Akad. v. Wetenseh., Amsterdam*, 50:302–314, 1947.

[17] J. Haantjes. A characteristic local property of geodesics in certain spaces. *Proc. Kon. Ned. Akad. v. Wetenseh., Amsterdam*, 54:66–73, 1951.

[18] Q. Han and J.-X. Hong. *Isometric Embedding of Riemannian Manifolds in Euclidean Spaces*. Mathematical Surveys and Monographs, **130**, AMS Providence, RI, 2006.

[19] M. Jin, J. Kim, and X. D. Gu. Discrete surface ricci flow: Theory and applications. In *LNCS 4647: Mathematics of Surfaces, Springer-Verlag, Berlin*, pages 209–232. 2007.

[20] W. A. Kirk. On curvature of a metric space at a point. *Pacific J. Math.*, 14:195–198, 1964.

[21] G. Lerman and J. T. Whitehouse. On d-dimensional d-semimetrics and simplex-type inequalities for high-dimensional sine functions. *Journal of Approximation Theory*, 156(1):52–81, 2009.

---

[9] but not piecewise Euclidean.

[22] K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.

[23] J. R. Munkres. *Elementary Differential Topology (rev. ed.)*. Princeton University Press, Princeton, N.J., 1966.

[24] H. Pajot. Analytic capacity, rectificability, menger curvature and the cauchy integral. In *LNM 1799, Springer-Verlag, Berlin*, pages 302–314. 2002.

[25] C. Plaut. Metric spaces of curvature $\geq k$. In *Handbook of Geometric Topology (Daverman, R. J. and Sher, R. B., editors)*, pages 819–898. 2002.

[26] M. Sageev. *CAT(0)* cube complexes and groups. *PCMI Lecture Notes*, pages 1–16, 2013. to appear.

[27] E. Saucan. On a construction of burago and zalgaller. *The Asian Journal of Mathematics*, 16(4):587–606, 2012.

[28] E. Saucan. A metric ricci flow for surfaces and its applications. *Geometry, Imaging and Computing*, 1(2):259–301, 2014.

[29] E. Saucan and E. Appleboim. Curvature based clustering for dna microarray data analysis. In *LNCS 3523, Springer-Verlag, Berlin*, pages 405–412. 2005.

[30] E. Saucan and E. Appleboim. Metric methods in surface triangulation. In *LNCS 5654, Springer-Verlag, Berlin*, pages 335–355. 2009.

[31] P. Shvartsman. Sobolev l_p^2-functions on closed subsets of $\mathbb{R}^2$. *arXiv:1210.0590*, 1(8):to appear, 2012.

[32] D. A. Stone. A combinatorial analogue of a theorem of myers. *Illinois J. Math.*, 20(1):12–21, 1976.

[33] D. A. Stone. Correction to my paper: "a combinatorial analogue of a theorem of myers". *Illinois J. Math.*, 20(1):551–554, 1976.

[34] D. A. Stone. Geodesics in piecewise linear manifolds. *Trans. Amer. Math. Soc.*, 215(1):1–44, 1976.

[35] A. Wald. Sur la courbure des surfaces. *C. R. Acad. Sci. Paris*, 201(8):918–920, 1935.

[36] A. Wald. Begreudeung einer koordinatenlosen differentialgeometrie der flächen. *Ergebnisse e. Mathem. Kolloquims, First Series*, 7(8):24–46, 1936.

Department of Mathematics, Technion, HAIFA 32000, and, Department of Mathematics and Computer Science, The Open University of Israel, RA'ANANA 43537, ISRAEL • semil@tx.technion.ac.il

Curvature in image and shape processing

Yonathan Aflalo, Anastasia Dubrovina, Ron Kimmel, and Aaron Wetzler

Abstract

The laplacian operator applied to the coordinates of a manifold provides the mean curvature vector. Manipulating the metric of the manifold or interpreting its coordinates in various ways provide useful tools for shape and image processing and representation. We will review some of these tools focusing on scale invariant geometry, curvature flow with respect to an embedding of the image manifold in a high dimensional space, and object segmentation by active contours defined via the shape laplacian operator. Such generalizations of the curvature vector and its numerical approximation as part of an image flow or triangulated shape representation, demonstrate the omnipresence of this operator and its usefulness in imaging sciences.

1. Introduction

The total variation [25] image selective smoothing filter, the Beltrami flow for color image processing [28, 24, 32] and its close relative the bilateral filter [30, 27], the affine invariant flow for images [10, 3] are all reincarnations of considering the heat flow defined by second derivative of a manifold with respect to its associated arc length. Strictly speaking, from a differential geometry point of view, the laplacian operator does not always defines the curvature, especially when non-trivial invariants are involved. Yet, this geometric structure has been found to be very practical in shape and image processing.

At the other end, the laplacian operator can be decomposed into its eigenfunctions and eigenvalues, which provide a generalization of the classical Fourier decomposition to non-trivial manifolds and non-regular parameterizations. Such a bases were introduced into the shape analysis arena in the last decade, see for example [18, 19]. The resulting eigenfunctions are sometimes referred to as the natural basis, as it can be shown to be the optimal basis for representing smooth functions on the given manifold [1]. Here, we will review several applications of the laplace Beltrami operator (LBO) in which we allow ourself to define the metric according to the problem at hand. The resulting heat operator or natural basis by decomposition of the LBO would allow us to process images and shapes with the same framework. In a sense, this paper is more a pedagogical overview rather than a specific contribution. To that end, let us start with a simple example of how a new definition of a pseudo-metric could lead to a scale invariant geometry for shapes.

2. On geometries and natural eigenbases

Sochen suggested to plugged invariant metrics into the definition of the laplacian in order to extract corresponding invariant eigenfunctions and eigenvalues for image processing [29]. Later on, Raviv et al. [23] used this observation to construct an equi-affine invariant pseudo metric for surfaces, while two of us and Raviv [2] introduced a scale invariant geometry and produced a corresponding natural basis for shapes. To exemplify the properties of such a methodology,
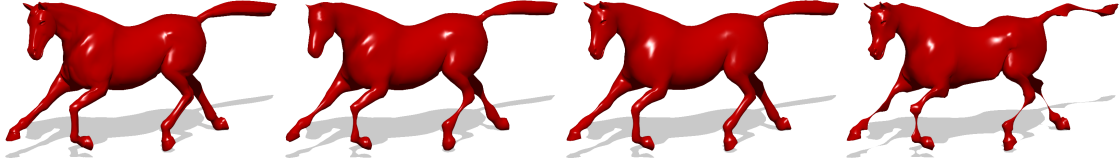
Figure 2.1: From left to right: A horse, Coordinates projected to the first 300 eigenfunctions (regular metric), Coordinates projected to the first 300 eigenfunctions (mixed metric). Coordinates projected to the first 300 eigenfunctions (scale invariant metric).

here, we start from an arc length construction followed by a metric definition and decompose the corresponding LBO into a natural basis structure that we then apply to shape representation.

Let $S(u, v) : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$ be a parametrized surface embedded in $\mathbb{R}^3$. Next, let $C(p) : [0, 1] \to S$ be a parametrized curve in S, for which we can define the total arc length to be

$$(2.1) \quad \text{length}(C) = \int_0^1 |C_p| dp = \int_0^1 |S_u u_p + S_v v_p| dp = \int_0^1 \sqrt{|S_u|^2 u_p^2 + 2\langle S_u, S_v \rangle u_p v_p + |S_v|^2 v_p^2} dp.$$

Denoting the metric coefficients $g_{ij} = \langle S_i, S_j \rangle$ where $i, j \in \{u, v\}$ we readily have the arc length on the surface defined by $ds^2 = \sum_{i=\{u,v\}} \sum_{j=\{u,v\}} g_{ij} di dj$. Using Einstein summation convention, the Laplace-Beltrami operator is given by

$$(2.2) \quad \Delta_g \equiv \frac{1}{\sqrt{g}} \partial_i \sqrt{g} g^{ij} \partial_j,$$

where g is the determinant of the metric (g), the inverse metric coefficients are $g^{ij} = \left((g)^{-1}\right)_{i,j}$, and ∂_i is the derivative with respect to the ith coordinate, u or v in our case.

The operator Δ_g admits a spectral decomposition with an orthogonal eigenbasis $\{\phi_k\}$ and a set of corresponding eigenvalues $\{\lambda_k\}$ defined by

$$(2.3) \quad \Delta_g \phi_k = \lambda_k \phi_k, \quad \langle \phi_k, \phi_k \rangle_g = 1, \quad \text{and} \quad \langle \phi_k, \phi_l \rangle_g = 0, \ \forall k \neq l.$$

The choice of an invariant metric is obviously important in the context of shape representation, alignment, and matching. To that end, various distances have been proposed, such as Euclidean [12, 5], geodesic [16, 21, 17, 20], diffusion [4, 14, 9], and affine invariant versions thereof [23]. A scale invariant geometry which is also intrinsic (embedding invariant) was introduced in [2] by which the regular metric is multiplied by the Gaussian curvature,

$$(2.4) \quad \tilde{g}_{ij} = |K| g_{ij} = |K| \langle S_i, S_j \rangle.$$

Fig. 2.1 depicts the effect of representing a shape's coordinates projected to the natural basis with a regular metric compared to the scale invariant one. The idea is to use just the first few eigenfunctions as an approximation for smooth functions on the manifold treating the coordinates as such. While the regular natural basis captures the global structure of the surface, as expected, the scale invariant basis concentrates at the fine features with effective curvature. Fig. 2.2 presents an application of the generalized multidimensional scaling algorithm [8] using the scale invariant diffusion distance to extract the correspondence between an armadillo and its deformed version In these examples the LBO produced a proper basis to represent and match shapes subject to a specific yet general enough deformations. Next we will show how the LBO can be applied as a non-local heat operator acting as a powerful selective smoothing filter in image denoising.

3. BELTRAMI SELECTIVE SMOOTHING IN PATCH SPACE

An image $I(x, y)$ can be thought of as a two dimensional surface embedded in the hybrid spatial-intensity space. The laplacian operator of the image manifold can then be used to define a heat flow applied as a filter to the image as initial condition

$$(3.1) \quad I_t = \Delta_g I, \quad \text{given} \ I(x, y; 0) = I_0.$$

The time t such a flow acts on the image should somehow reflect the amount of noise in the image, while the metric g could be the image metric. The hope is that such a curvature flow
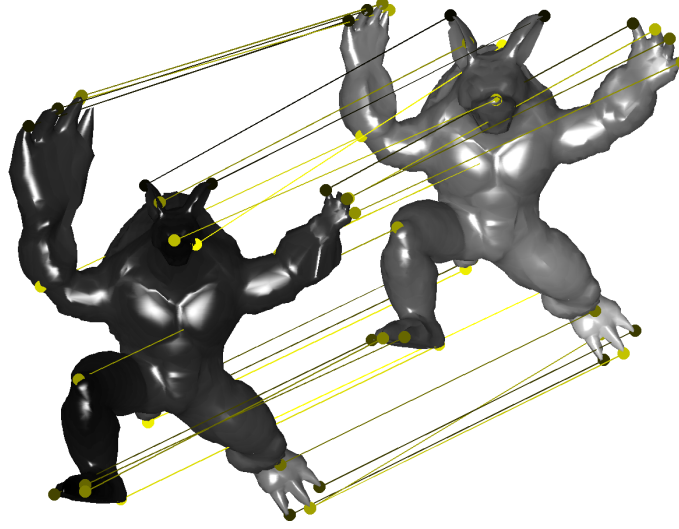
Figure 2.2: Matching an Armadillo and its locally scaled version using GMDS with the scale invariant metric.

would selectively smooth the image in a desirable fashion. The Beltrami-flow [28] considers a color image as a mapping between the spatial domain to the spatial-spectral domain, that is $I(x,y) : \Omega \in \mathbb{R}^2 \to \mathbb{R}^5$. The coordinates in the embedding space are x, y, R, G, B where the last three coordinates stand for the color space Red, Green, and Blue, respectively. The resulting heat equation $I_t = \Delta_g I$ defined by the metric g is usually derived from an infinitesimal arc length such as $ds^2 = dx^2 + dy^2 + dR^2 + dG^2 + dB^2$.

One non-differential version of the Beltrami flow is known as the bilateral filter [31], in which the short time kernel of the Beltrami flow is replaced by $\delta(s, s')^2 = \delta(x, x')^2 + \delta(y, y')^2 + \delta(R, R')^2 + \delta(G, G')^2 + \delta(B, B')^2$, where $\delta(x, x')^2 = (x - x')^2$. Roughly speaking, instead of convolving with the continuous version $e^{-(\int_s^{s'} ds)^2}$ in the Beltrami flow, the bilateral filter applies $e^{-\delta(s,s')^2}$ as a selective smoothing kernel.

Roussos and Maragos [24] followed by [32] extended the Beltrami flow approach by treating images as two dimensional manifolds embedded in *patch-space*. The superior denoising results of these methods coupled with their computational efficiency indicated that the Beltrami flow in its general form is well suited for dealing with image denoising. An example of the selective smoothing property of the Beltrami flow in patch space is depicted by Fig. 3.1.

In order to simplify the construction of such filters consider a height profile I to be a two dimensional Riemannian manifold embedded in a higher dimensional space. We define the patch-space mapping $P : \Omega \in \mathbb{R}^2 \to S \in \mathbb{R}^{n(2w+1)^2+2}$ such that

$$(3.2) \qquad P(x,y) = \left(x, y, \left\{I_{i,j}^k\right\}\right),$$

for $i, j = -w, .., w, \quad k = 1, .., n$,where $w \in \mathbb{N}$ is the window size, n is the number of channels we use, and $\{I_{i,j}^k\}$ is the compact form for $\left\{I^k\left(x + i, y + j\right)\right\}$. For the case of a single height profile or a gray level image $n = 1$, where if we were provided with a set of registered scans of a particular surface or a color image, n could represent the number of scans or colors. The manifold P is equipped with a metric g, that defines an arc length

$$(3.3) \qquad ds^2 = (dx \ dy) \, (g) \left(\begin{array}{c} dx \\ dy \end{array} \right).$$

Specifically, the patch-space metric is given by application of chain rule $dI_{i,j}^k = \partial_x I_{i,j}^k dx + \partial_y I_{i,j}^k dy$ from which it follows that

$$(3.4) \qquad (g) = \left(\begin{array}{cc} 1 + \sum_{i,j,k} (\partial_x I_{i,j}^k)^2 & \sum_{i,j,k} \partial_x I_{i,j}^k \partial_y I_{i,j}^k \\ \sum_{i,j,k} \partial_x I_{i,j}^k \partial_y I_{i,j}^k & 1 + \sum_{i,j,k} (\partial_y I_{i,j}^k)^2 \end{array} \right).$$

133

Figure 3.1: Examples of Beltrami patch denoising for images. From top to bottom, left to right a) Noisy Barbara, $\sigma = 21$ b) Denoised image, PSNR = 28.4dB c) Noisy Peppers, $\sigma = 30$ d) Denoised image, PSNR = 28.0dB

This metric allows us to measure distances and areas on the manifold P using the coordinates of Ω. For example, the area of the embedding into a Euclidean space is given by

$$(3.5) \qquad \iint \sqrt{\det(g)}dxdy.$$

We minimize Eq. (3.5) using variational calculus and arrive at the EL equation

$$(3.6) \qquad \Delta_g I_{i,j}^k = \frac{1}{\sqrt{g}}\mathrm{div}\big(\sqrt{g}(g)^{-1}\nabla I_{i,j}^k\big) = 0.$$

The resulting operator is indeed the LBO, and the image can thus be filtered by the corresponding Beltrami flow in patch space

$$(3.7) \qquad I_t = \Delta_g I.$$

Point clouds and meshes are often assumed to have noise which is Gaussian and appears as an offset along the normal direction of the surface. For that kind of noise the mean curvature flow could produce a reasonable filter. Yet, the noise model for range scanners exhibits offsets in the direction of the camera's focal point. Moreover, the noise is seldom Gaussian and usually includes regions with missing data. This presents a challenge to state of the art denoising algorithms that are usually tuned for optimal removal of additive white noise. One of the popular denoising methods is the BM3D [15] to which we compare the Beltrami patch denoising for a depth profile obtained from a real scanner, as seen in Fig. 3.2. It appears that the method is able to selectively smooth the data while preserving the important features. The usage of the LBO as a non-local denoising filter is not our final stop. Next, we show how it can be used to formulate the geodesic active contour model in a new and novel formulation.

4. An LBO perspective on geodesic active contours

This section links between the Laplace-Beltrami operator, the associated heat flow towards a minimal surface, and the Geodesic Active Contours (GAC) model for image segmentation [11]. Related efforts include Bresson's *et al.* [7], expressing the GAC flow as a minimizer of the *weighted Polyakov action*, and the formulation by Bogdanova *et al.* [6]. Sochen *et al.* [26] reviewed the
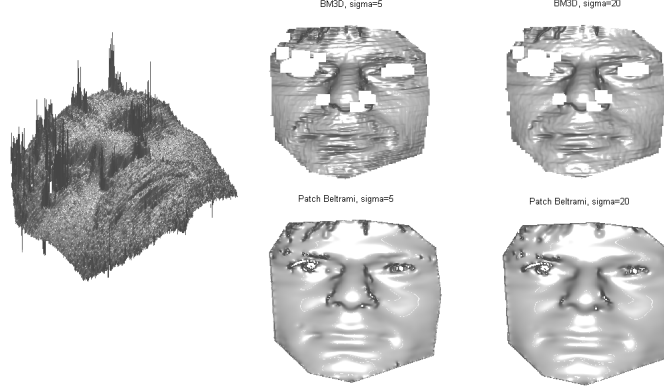
Figure 3.2: Far left: Original noisy scan. Top row: BM3D results for $\sigma = 5$ and $\sigma = 20$. Bottom row: Beltrami patch denoising for $\sigma = 5$ and $\sigma = 20$.

image filtering problem, exploring the relation between the PDE geometric approaches, derived by minimizing the Polyakov action with an appropriate metric, and non-linear robust-statistics filters. We start by reviewing the geodesic active contour model and the evolution towards generalized minimal surface.

4.1. **Geodesic active contours.** The geodesic active contour is a geometric-variational model for boundary detection, integration, and object segmentation in images. In [13] it was shown that contours extracted by this model are minimal geodesics in a Riemannian space whose metric is defined by the image intensity $I(x, y)$. We search for curves $C(p) : [0, 1] \to \mathbb{R}^2$ that minimize the following weighted Euclidean arc length

$$(4.1) \qquad \int f\left(|\nabla I\left(C(p)\right)|\right) |C'(p)|\, dp = \int f\left(|\nabla I\left(C(s)\right)|\right) ds,$$

in a Riemannian space equipped with the metric tensor $f\left(|\nabla I\left(C\right)|\right) \delta_{ij}$. The function $f(x, y)$ is an edge indicator function, for example $f(x, y) = (1 + |\nabla I|)^{-1}$, designed to stop the active contour when it arrives at the object's boundary. The Euclidean arc length is denoted by s, where $ds = |C'(p)|dp$. Caselles *et al.* used a gradient-descent method to minimize the weighted arc length (4.1), to obtain the following curve evolution equation

$$(4.2) \qquad \frac{\partial C(t)}{\partial t} = \left(f\kappa - \nabla f \cdot \vec{N}\right) \vec{N},$$

where $\kappa\vec{N} = C_{ss}$, with κ is the curvature of C, and $\vec{N}$ its unit inward pointing normal.

The level set formulation [22] of the geodesic problem (4.2) is given by

$$(4.3) \qquad \frac{\partial u}{\partial t} = |\nabla u|\operatorname{div}\left(f\frac{\nabla u}{|\nabla u|}\right) = f\,|\nabla u|\operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) + \nabla f \cdot \nabla u,$$

where $u(x, y; t)$ is an implicit representation, often referred to as a level set function, of the evolving contour $C(p; t)$. That is, $C(t) = \{(x, y) : u(x, y; t) = 0\}$, and $\kappa = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)$ is the curvature of the level sets of the function u. The level set evolution formulation in Eq. (4.3) can be interpreted as a generalized minimal surface flow. To that end, we apply the methodology provided by Sochen *et al.* in [28] as described next.

4.2. **Generalized minimal surface flow.** Let us treat the level set function $u(x, y)$ as a two-dimensional surface embedded in a 3-dimensional space. It can be defined by the map $\mathbf{X} : \Omega \in \mathbb{R}^2 \to M \in \mathbb{R}^3$, where Ω denotes a 2D coordinates manifold (σ_1, σ_2), and M is the embedding space. Explicitly, $\mathbf{X}$ is written as $\mathbf{X} = \left(X^1(\sigma^1, \sigma^2), X^2(\sigma^1, \sigma^2), X^3(\sigma^1, \sigma^2)\right)$. Both manifolds Ω and M are equipped with metric tensors, $g_{\mu\nu}(\sigma^1, \sigma^2)$ and $h_{ij}(x^1, x^2, x^3)$, respectively. The map $\mathbf{X}$ and the metric h_{ij} can be used to construct the metric on Ω

$$(4.4) \qquad g_{\mu\nu}(\sigma^1, \sigma^2) = h_{ij}(\mathbf{X})\partial_\mu X^i \partial_\nu X^j,$$

where we used Einstein's summation convention. Next, the following functional can be associated with the map $\mathbf{X} : \Omega \to M$,

$$(4.5) \qquad S\left[X^i, g_{\mu\nu}, h_{ij}\right] = \int d^m \sigma \sqrt{g} g^{\mu\nu} \partial_\mu X^i \partial_\nu X^j h_{ij}(\mathbf{X}),$$

where $g^{\mu\nu}$ is the inverse of the metric $g_{\mu\nu}$ (that is $g^{\mu\gamma} g_{\gamma\nu} = \delta_{\mu\nu}$), and g is the determinant of $(g_{\mu\nu})$. This functional is known as Polyakov action, and it can be viewed as a generalized area measure. In fact, we already used a particular instance of the Polyakov action in Section 3, Eq. (3.5).

The minimal map (embedding) $\mathbf{X}$ can be obtained by steepest-descent. The gradient of the Polyakov action with respect to the embedding is

$$(4.6) \qquad -\frac{1}{2\sqrt{g}} h^{il} \frac{\delta S}{\delta X^l} = \frac{1}{\sqrt{g}} \partial_\mu \left(\sqrt{g} g^{\mu\nu} \partial_\nu X^i \right) + \Gamma^i_{jk} \partial_\mu X^j \partial_\nu X^k g^{\mu\nu}.$$

In order to find the minimal embedding, Sochen *et al.* [28] used the gradient descent flow

$$(4.7) \qquad X^i_t = -\frac{1}{2\sqrt{g}} h^{il} \frac{\delta S}{\delta X^l} = \Delta_g X^i + \Gamma^i_{jk} \partial_\mu X^j \partial_\nu X^k g^{\mu\nu}.$$

Note, that the gradient (4.6) is obtained by multiplying the Euler-Lagrange equations of (4.5) by a strictly positive function and a positive definite matrix, that will be referred to as a *pre-factor*. It could also be viewed as an indication of the metric of the variational inner product by which the gradient descent is defined. It provides a geometric parameterization-invariant flow with the same minimum. The pre-factor required to produce the GAC flow is different, stemming from the different geometry of the problem. The second term at the right hand side of Eq. (4.6) involves the Levi-Civita connection coefficients Γ^i_{jk}, that depict the geometry of the embedding space. When $M = \mathbb{R}^3$ with Euclidean metric, $h_{ij} = \delta_{ij}$, the second term vanishes, and the flow becomes $X_t = \Delta_g X$, as seen in the previous sections.

4.3. **Back to GAC: level set formulation as a flow toward minimal surface.** Next, we show that the level set formulation for geodesic active contour evolution in Eq. (4.3) is obtained by minimizing a generalized area measure. First, let us select $\mathbf{X}$ that maps the plane $(\sigma^1 = x, \sigma^2 = y)$ to a 3D Euclidean space, such that

$$(4.8) \qquad \mathbf{X} = (x, y, u(x,y)).$$

The functional we would like to study is

$$(4.9) \qquad S = \iint f(|\nabla I(x,y)|) \sqrt{1 + |\nabla u|^2} dx dy,$$

This is Polyakov action obtained by choosing the following metric tensors for the parameter and the embedding spaces Ω and M, respectively

$$(4.10) \qquad \begin{aligned} g_{\mu\nu} &= f(\sigma^2, \sigma^2) \left(\partial_\mu X \cdot \partial_\nu X \right), \\ h_{ij} &= f(x^1, x^2) \delta_{ij}. \end{aligned}$$

Since $g_{\mu\nu}$ and h_{ij} are legitimate metric tensors, and as $(\sigma^2, \sigma^2) = (x, y)$ and $(x^1, x^2, x^3) = (x, y, z)$, Eq. (4.4) holds.

The metric tensor $g_{\mu\nu}$, written in a matrix form becomes

$$(4.11) \qquad (g_{\mu\nu}) = f \begin{pmatrix} 1 + u_x^2 & u_x u_y \\ u_x u_y & 1 + u_y^2 \end{pmatrix}.$$

The metric determinant is $g = \det(g_{\mu\nu}) = f^2 \left(1 + |\nabla u|^2 \right)$ and the inverse of the metric is

$$(4.12) \qquad (g_{\mu\nu})^{-1} = (g^{\mu\nu}) = \frac{f}{g} \begin{pmatrix} 1 + u_y^2 & -u_x u_y \\ -u_x u_y & 1 + u_x^2 \end{pmatrix}.$$

Next, we use Eq. (4.6) to obtain the gradient-descent flow for the level set function component of $\mathbf{X}$, namely $X^3 = u(x,y)$,

$$(4.13) \qquad u_t = X^3_t = \frac{1}{\sqrt{g}} \partial_\mu \left(\sqrt{g} g^{\mu\nu} \partial_\nu u \right) + \Gamma^3_{jk} \partial_\mu X^j \partial_\nu X^k g^{\mu\nu}.$$
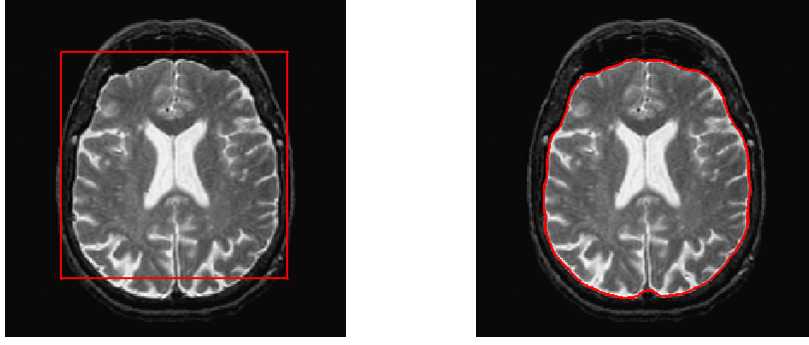
Figure 4.1: Image segmentation using Geodesic Active Contours: the initial contour (left) and the final segmentation (right).

Substituting the expressions for $\sqrt{g}$ and $g^{\mu\nu}$ into the first term of the right-hand side of the flow in Eq. (4.13) produces

$$(4.14) \qquad \frac{1}{\sqrt{g}}\partial_\mu\left(\sqrt{g}g^{\mu\nu}\partial_\nu u\right) = \frac{1}{f\sqrt{1+|\nabla u|^2}}\mathrm{div}\left(\frac{\nabla u}{\sqrt{1+|\nabla u|^2}}\right).$$

In order to compute the second term on the r.h.s. of Eq. (4.13) we need to first find an expression for the Levi-Civita connection coefficients Γ^3_{jk}. For the metric h_{ij} defined in Eq. (4.10)

$$
\begin{aligned}
\Gamma^i_{jk} &= \frac{1}{2}h^{il}\left(\partial_j h_{lk} + \partial_k h_{jl} - \partial_l h_{jk}\right) \\
&= \sum_l \frac{1}{2}\frac{1}{f}\delta_{il}\left(\partial_j\left(f\delta_{lk}\right) + \partial_k\left(f\delta_{jl}\right) - \partial_l\left(f\delta_{jk}\right)\right) \\
&= \frac{1}{2}\frac{1}{f}\left(\delta_{ik}\partial_j f + \delta_{ji}\partial_k f - \delta_{jk}\partial_i f\right).
\end{aligned}
$$

(4.15)

Therefore,

$$(4.16) \qquad \Gamma^3_{jk} = \frac{1}{2f}\left(\delta_{3k}\partial_j f + \delta_{j3}\partial_k f\right),$$

or, in a matrix form,

$$(4.17) \qquad \Gamma^3 = \frac{1}{2f}\begin{pmatrix} 0 & 0 & f_x \\ 0 & 0 & f_y \\ f_x & f_y & 0 \end{pmatrix}.$$

Finally, the second term of the flow from Eq. (4.13) becomes

$$(4.18) \qquad \Gamma^3_{jk}\partial_\mu X^j\partial_\nu X^k g^{\mu\nu} = \frac{\nabla f\cdot\nabla u}{f^2\left(1+|\nabla u|^2\right)}.$$

Using Eq. (4.14) and Eq. (4.18) we obtain the level set evolution

$$(4.19) \qquad u_t = \frac{1}{f\sqrt{1+|\nabla u|^2}}\mathrm{div}\left(\frac{\nabla u}{\sqrt{1+|\nabla u|^2}}\right) + \frac{\nabla f\cdot\nabla u}{f^2\left(1+|\nabla u|^2\right)}.$$

Finally, we use our freedom of parametrization and multiply the above evolution by the pre-factor of $f^2\left(1+|\nabla u|^2\right)$, to obtain

$$(4.20) \qquad u_t = \mathrm{div}\left(f\frac{\nabla u}{\sqrt{1+|\nabla u|^2}}\right)\sqrt{1+|\nabla u|^2}.$$

Note, that up to the additional constant 1, this formulation aligns with the level set formulation of the geodesic active contour model given in Eq. (4.3). Since the surface definition in Eq. (4.8) is arbitrary, we can choose the aspect ratio between du and dx, dy to be as large as we want. Thus, the constant 1 can be regarded as an $\varepsilon \to 0$ that vanishes upon the right selection of this aspect ratio. We thereby demonstrated that the geodesic active contour model in its level set

formulation, can also be viewed as a surface that minimizes the Polyakov action with a specific selection of metric tensors for the parametrization and the embedding space. The minimizer is obtained by application of the LBO, yet again in a gradient descent heat flow fashion.

5. Conclusions

The Laplace Beltrami operator has been shown to provide a modeling construction approach which is useful for shape and image selective smoothing, for shape matching, and for object segmentation in images. We allowed ourselves to select the relevant metric and the coordinates upon which the operator is acting according to the problem at hand. The freedom of these settings allowed us to link between state of the art solutions for shape and image processing and analysis. We expect these observations and tool design methodology to pave the way for new solutions of novel problems in the domain of imaging sciences.

6. Acknowledgements

References

[1] Yonathan Aflalo and Ron Kimmel. Spectral multidimensional scaling. *Proceedings of the National Academy of Sciences*, 110(45):18052–18057, 2013.

[2] Yonathan. Aflalo, Ron Kimmel, and Dan Raviv. Scale invariant geometry for nonrigid shapes. *SIAM Journal on Imaging Sciences*, 6(3):1579–1597, 2013.

[3] Luis Alvarez, Frédéric Guichard, Pierre-Louis Lions, and Jean-Michel Morel. Axioms and fundamental equations of image processing. *Archive for Rational Mechanics and Analysis*, 123(3):199–257, 1993.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.*, 15(6):1373–1396, 2003.

[5] P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[6] I. Bogdanova, X. Bresson, J. P. Thiran, and P. Vandergheynst. Scale space analysis and active contours for omnidirectional images. *Image Processing, IEEE Transactions on*, 16(7):1888–1901, 2007.

[7] X. Bresson, P. Vandergheynst, and J. P. Thiran. Multiscale active contours. *Scale Space and PDE Methods in Computer Vision*, pages 167–178, 2005.

[8] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. *Proceedings of the National Academy of Science*, pages 1168–1172, 2006.

[9] A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro. A Gromov-Hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *International Journal of Computer Vision*, 89(2-3):266–286, 2010.

[10] A.M. Bruckstein, G. Sapiro, and D. Shaked. *Affine-invariant Evolutions of Planar Polygons*. CIS report: ham-@Merkāz le-Ma?arākôt Nevônôt. 1992.

[11] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, February 1997.

[12] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 3, pages 2724 –2729, 1991.

[13] L. D. Cohen and R. Kimmel. Global minimum for active contour models: A minimal path approach. *International Journal of Computer Vision*, 24(1):57–78, 1997.

[14] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5 – 30, 2006. Special Issue: Diffusion Maps and Wavelets.

[15] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, pages 2080–2095, 2007.

[16] A. Elad (Elbaz) and R. Kimmel. On bending invariant signatures for surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 25(10):1285–1295, 2003.

[17] M. Hilaga, Y. Shinagawa, T. Kohmura, and T.L. Kunii. Topology matching for fully automatic similarity estimation of 3D shapes. In *ACM SIGGRAPH 2001*, Los Angeles, CA, 12-17 August 2001.

[18] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, pages 279–286, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[19] B. Levy. Laplace-beltrami eigenfunctions towards an algorithm that "understands" geometry. *Conference on Shape Modeling and Applications, 2006. SMI 2006. IEEE International*, pages 13–13, 2006.

[20] F. Mémoli and G. Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Foundations of Computational Mathematics*, 5(3):313–347, 2005.

[21] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transactions on Graphics*, 21(4):807–832, 2002.

[22] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[23] D. Raviv, A. M. Bronstein, M. M. Bronstein, R. Kimmel, and N. Sochen. Affine-invariant geodesic geometry of deformable 3d shapes. *Computers & Graphics*, 35(3):692–697, 2011.

[24] A. Roussos and P. Maragos. Tensor-based image diffusions derived from generalizations of the total variation and Beltrami functionals. In *ICIP*, September 2010.

[25] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60(1-4):259–268, November 1992.

[26] N. Sochen, R. Kimmel, and A. M. Bruckstein. Diffusions and confusions in signal and image processing. *Journal of Mathematical Imaging and Vision*, 14(3):195–209, 2001.

[27] N. Sochen, R. Kimmel, and A.M. Bruckstein. Diffusions and confusions in signal and image processing. *Journal of Mathematical Imaging and Vision*, 14(3):195–209, 2001.

[28] N. Sochen, R. Kimmel, and R. Malladi. A general framework for low level vision. *IEEE Trans. on Image Processing*, pages 310–318, 1998.

[29] Nir A. Sochen. Stochastic processes in vision: From langevin to beltrami. *Computer Vision, IEEE International Conference on*, 1:288, 2001.

[30] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proc. IEEE ICCV*, pages 836–846, 1998.

[31] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 839–, Washington, DC, USA, 1998. IEEE Computer Society.

[32] A. Wetzler and R. Kimmel. Efficient beltrami flow in patch-space. In *Proceedings of the 3rd International Conference on Scale Space and Variational Methods in Computer Vision 2011*, volume 6667, pages 134–143. Springer, 2011.

GIP Lab, Technion, Haifa 32000, Israel • yaflalo@cs.technion.ac.il • nastyad@cs.technion.ac.il • ron@cs.technion.ac.il • twerd@cs.technion.ac.il

Geometric Aspects of the Space of Triangulations

Pooran MEMARI

Abstract

These are the notes of my talk presented in the colloquium on discrete curvature at the CIRM, in Luminy (France) on November 21st, 2013, in which we study the space of triangulations from a purely geometric point of view and revisit the results presented in [21] and [20] **(joint works with Patrick Mullen, Fernando De Goes and Mathieu Desbrun)**. Motivated by practical numerical issues in a number of modeling and simulation problems, we first introduce the notion of a *compatible dual complex* (made out of convex cells) to a primal triangulation, such that a simplicial mesh and its compatible dual complex form what we call a *primal-dual triangulation*. Using algebraic and computational geometry results, we show that for simply connected domains, compatible dual complexes exist only for a particular type of triangulation known as weakly regular. We also demonstrate that the entire space of primal-dual triangulations, which extends the well known (weighted) Delaunay/Voronoi duality, has a convenient, geometric parameterization. We finally discuss how this parameterization may play an important role in discrete optimization problems such as optimal mesh generation, as it allows us to easily explore the space of primal-dual structures along with some important subspaces.

1. INTRODUCTION

Mesh generation traditionally aims at tiling a bounded spatial domain with simplices (triangles in 2D, tetrahedra in 3D) so that any two of these simplices are either disjoint or sharing a lower dimensional face. The resulting triangulation provides a discretization of space through both its primal (simplicial) elements *and* its dual (cell) elements. Both types of element are crucial to a variety of numerical techniques, finite element (FE) and finite volume (FV) methods being arguably the most widely used in computational science. To ensure numerical accuracy and efficiency, specific requirements on the size and shape of the primal (typically for FE) or the dual elements (typically for FV) in the mesh are often sought after.

Towards Generalized Primal/Dual Meshes. A growing trend in numerical simulation is the simultaneous use of primal *and* dual meshes: Petrov-Galerkin finite-element/finite-volume methods (FE/FVM, [3, 19, 24]) and exterior calculus based methods [4, 6, 13] use the ability to store quantities on both primal and dual elements to enforce (co)homological relationships in, e.g., Hodge theory. The choice of the dual, defined by the location of the dual vertices, is however not specified a priori. A very common dual to a triangulation in $\mathbb{R}^d$ is the cell complex which uses the circumcenters of each d-simplex as dual vertices. The barycentric dual, for which barycenters are used instead of circumcenters, is used for certain finite-volume computations, but it fails to satisfy both the orthogonality and the convexity conditions on general triangulations.

While the circumcentric Delaunay-Voronoi duality [25, 8] is one of the cornerstones of meshing methods and, as such, has been extensively used in diverse fields, more general dualities are often desired. Building on a number of results in algebraic and computational geometry, in [20] we present a more general primal-dual pairs of complexes, *primal-dual triangulations*, that we briefly describe in the first sections of this note.

In the last part of this note, we show how the Weighted-Delaunay/Laguerre duality could help to provide lower error bounds on numerical computations [21]. While most previous meshing methods focused on designing well-shaped primal triangulations *or* dual complexes, we provide a unifying approach to mesh quality based on the placement of primal *and* orthogonal dual elements with respect to each other. In an effort to provide meshes most appropriate for fast, yet reliable computations, we propose functionals on primal-dual mesh pairs that offer formal bounds on the numerical error induced by the use of diagonal Hodge stars. We then demonstrate that meshes that minimize our functionals have desirable geometric and numerical properties. These resulting Hodge-optimized meshes offer a much-needed alternative to the traditional use of barycentric or circumcentric duals in discrete computations. Finally, the resulting set of meshing tools we introduce has wide applications: even when a specific connectivity is needed, some of our contributions can be applied to increase numerical robustness and accuracy of basic operators.

2. Preliminaries

We start by reviewing important notions that we build upon and extend in subsequent sections.

2.1. **Complex, Subdivision, and Triangulation.** A **cell complex** in $\mathbb{R}^d$ is a set K of convex polyhedra (called cells) satisfying two conditions:

(1) Every face of a cell in K is also a cell in K, and
(2) If C and C' are cells in K, their intersection is a common face of both.

A **simplicial complex** is a cell complex whose cells are all simplices. The **body** $|K|$ of a complex K is the union of all its cells. When a subset P of $\mathbb{R}^d$ is the body of a complex K, then K is said to be a **subdivision** of P; if, in addition, K is a simplicial complex, then K is said to be a **triangulation** of P. For a set X of points in $\mathbb{R}^d$, a triangulation of X is a simplicial complex K for which each vertex of K is in X. In that case the body of K is the convex hull of X. Let us note that the triangulations we consider in this work, usually coming from a point set, they partition a simply connected domain in $\mathbb{R}^d$ (corresponding to the convex hull of the point set).

Also, in the definition of a triangulation of X, we do not require all the points of X to be used as vertices; a point $\mathbf{x_i} \in X$ is called *hidden* if it is not used in the triangulation. A triangulation with no hidden points is called a *full* triangulation.

2.2. **Triangulations in $\mathbb{R}^d$ through Lifting in $\mathbb{R}^{d+1}$.** Let $\mathbf{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ be a set of points in $\mathbb{R}^d$. A simple way of constructing a triangulation of $\mathbf{X}$ is through the following *lifting procedure*: take an arbitrary function $L : \mathbf{X} \longrightarrow \mathbb{R}$ called the *lifting function*; consider the points $(\mathbf{x_i}, L(\mathbf{x_i})) \in \mathbb{R}^{d+1}$, i.e., the points of $\mathbf{X}$ *lifted* onto the graph of L; in the space $\mathbb{R}^{d+1}$, consider Conv(L) the convex hull of vertical rays $\{(x_i, l) | \ l \geq L(\mathbf{x_i}), l \in \mathbb{R}, x_i \in X\}$; the bounded faces of Conv(L), i.e. faces which do not contain vertical half lines, form the **lower envelope** of the lifting L. If the function L is generic (see [11] Chap. 7), the orthogonal projection (onto the first d coordinates) of the lower envelope of L partitions the convex hull of $\mathbf{X}$ and produces a triangulation of $\mathbf{X}$.

It is clear that the above lifting procedure may produce triangulations for which not all points of $\mathbf{X}$ are vertices. A triangulation of a set $\mathbf{X}$ of points obtained through lifting is full (i.e., has no hidden points) if and only if all the points $(\mathbf{x_i}, L(\mathbf{x_i}))$ lie on the **lower envelope of** L (or, in other words, if function L can be extended, through linear interpolation in the triangles, to a **convex** piecewise-linear function).
Regular Triangulations : A triangulation obtained by orthogonally projecting the lower envelope of a lifting of $\mathbf{X}$ in $\mathbb{R}^{d+1}$ (onto the first d coordinates) is called a *regular triangulation* ([29], Definition 5.3).

2.3. **Weighted Delaunay Triangulations.** A special choice for the lifting function produces the well-known and widely-used Delaunay triangulation (see [26, 22] for properties of Delaunay triangulations, and [25] for numerous applications). Indeed, let X be a set of points in $\mathbb{R}^d$. Consider the *lifting* of the points in X onto the surface of the paraboloid $h(\mathbf{x}) = \|\mathbf{x}\|^2$ in $\mathbb{R}^{d+1}$; i.e., each $\mathbf{x_i} = (a_1, \ldots, a_d) \in X$ gets mapped to $(\mathbf{x_i}, h_i) \in \mathbb{R}^{d+1}$ with $h_i = \|\mathbf{x_i}\|^2 = a_1^2 + \cdots + a_d^2$. Then the orthogonal projection of the lower envelope of this lifting partitions the convex hull of $\mathbf{X}$ and produces a (full) triangulation coinciding with the **Delaunay triangulation** of $\mathbf{X}$.

A regular triangulation (of the convex hull) of a point set $\mathbf{X}$ can now be seen as a generalization of the Delaunay triangulation of $\mathbf{X}$ as follows. We first define a **weighted point set** as a set $(\mathbf{X}, W) = (\mathbf{x_1}, w_1), \ldots, (\mathbf{x_n}, w_n)$, where $\mathbf{X}$ is a set of points in $\mathbb{R}^d$, and $\{w_i\}_{i \in [1,\ldots,n]}$ are real numbers called weights. The **weighted Delaunay triangulation** of $(\mathbf{X}, W)$ is then the triangulation of $\mathbf{X}$ obtained by projecting the lower envelope of the points $(\mathbf{x_i}, \|\mathbf{x_i}\|^2 - w_i) \in \mathbb{R}^{d+1}$. Note that a weighted Delaunay triangulation can now have hidden points.

Notice also that given a lifting function L and its values $l_i = L(\mathbf{x_i})$ at the points of X, one can always define weights to be the difference between the paraboloid and the function L, $w_i = \|\mathbf{x_i}\|^2 - l_i$. We conclude that for simply connected domains (i.e. convex hull of $\mathbf{X}$), the notions of regular triangulations and weighted Delaunay triangulations are *equivalent*. Let us note that as it is shown in [5], this equivalence is no longer true for domains with non trivial topology.

2.4. **Generalized Voronoi Diagrams vs. Weighted Delaunay Triangulation.** Delaunay triangulations (resp., weighted Delaunay triangulations) can also be obtained (or defined) from their dual *Voronoi diagrams* (resp, *power diagrams*). Let $(\mathbf{X}, W) = \{(\mathbf{x_i}, w_i)\}_{i \in I}$ be a weighted point set in $\mathbb{R}^d$. The power of a point $\mathbf{x} \in \mathbb{R}^d$ with respect to a weighted point $(\mathbf{x_i}, w_i)$ (sometimes referred to as the Laguerre distance) is defined as $d^2(\mathbf{x}, \mathbf{x_i}) - w_i$, where $d(.,.)$ stands for the Euclidean distance. Using this power definition, to each x_i we associate its weighted Voronoi region $V(\mathbf{x_i}, w_i) = \{\mathbf{x} \in \mathbb{R}^d \mid d^2(\mathbf{x}, \mathbf{x_i}) - w_i \leq d^2(\mathbf{x}, \mathbf{x_j}) - w_j, \forall j\}$. The power diagram of $(\mathbf{X}, W)$ is the cell complex whose cells are the weighted Voronoi regions.

Note that when the weights are all equal, the power diagram coincides with the Euclidean Voronoi diagram of $\mathbf{X}$. Power diagrams are well known to be dual to weighted Delaunay triangulations, as we review next.

The **dual** of the power diagram of $(\mathbf{X}, W)$ is the weighted Delaunay triangulation of $(\mathbf{X}, W)$. This triangulation contains a k-simplex with vertices $\mathbf{x_{a_0}}, \mathbf{x_{a_1}}, \ldots, \mathbf{x_{a_k}}$ in $\mathbf{X}$ if and only if $V(x_{a_0}, w_{a_0}) \cap V(x_{a_1}, w_{a_1}) \cap \cdots \cap V(x_{a_k}, w_{a_k}) \neq \emptyset$, $\forall k \geq 0$. While many other generalization of Voronoi diagrams exist, they do not form straight-edge and convex polytopes, and are thus not relevant here.

3. Compatible Dual Complexes of Triangulations

We now show that the notion of mesh duality can be extended so that the dual complex is defined geometrically, and independently from the triangulation—while the combinatorial compatibility between the triangulation and its dual is maintained.

Definition 1 (Simple Cell Complex)**.** A cell complex K in $\mathbb{R}^d$ is called **simple** if every vertex of K is incident to $d + 1$ edges. K is called labeled if every d-dimensional cell of K is assigned a unique label; in this case, we write $K = \{C_1, \ldots, C_n\}$, where n is the number of d-dimensional cells of K, and C_i is the i-th d-dimensional cell.

Definition 2 (Compatible Dual Complex)**.** Let T be a triangulation of a set $\mathbf{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ of points in $\mathbb{R}^d$; and $K = \{C_{i_1}, \ldots, C_{i_n}\}$ be a labeled simple cell complex, i.e. there is a one-to-one correspondence between x_p and C_{i_p}. K is called a **compatible dual complex** of T if, for every pair of points $\mathbf{x_p}$ and $\mathbf{x_q}$ that are connected in T, C_{i_p} and C_{i_q} share a face.

This compatibility between K and T is purely combinatorial, i.e., it simply states that the connectivity between points induced by K coincides with the one induced by T. Notice that the cell C_{i_p} associated to the point $\mathbf{x_p}$, does not necessarily contain $\mathbf{x_p}$ in its interior. Moreover, the edge $[\mathbf{x_p}, \mathbf{x_q}]$ and its *dual* $C_{i_p} \cap C_{i_q}$ are not necessarily *orthogonal* to each other, unlike most conventional geometric dual structures. Consequently, we can generalize the notion of mesh duality through the following definition:

Definition 3 (Primal-Dual Triangulation (PDT))**.** A pair (T, K) is said to form a d-dimensional **primal-dual triangulation** if T is a triangulation in $\mathbb{R}^d$ and K is a compatible dual complex of T. If every edge $[\mathbf{x_p}, \mathbf{x_q}]$ and its *dual* $C_{i_p} \cap C_{i_q}$ are orthogonal to each other, the pair (T, K) is said to form an **orthogonal primal-dual triangulation**.
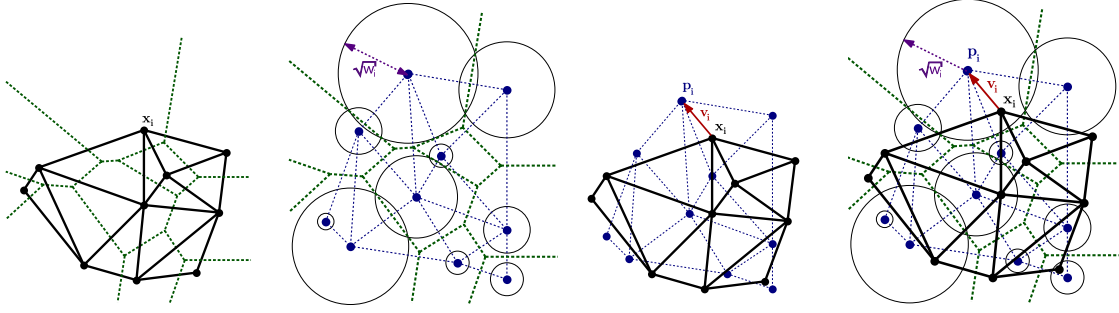
Figure 3.1: *Primal-Dual Triangulation: primal triangulation, dual complex, and combinatorially equivalent regular triangulation separately displayed for clarity.*

3.1. **Characterization of Primal-Dual Triangulations.** An immediate question is whether any triangulation can be part of a PDT. We first characterize the triangulations that admit a compatible dual complex through the following two definitions:

Definition 4 (Combinatorial Equivalence)**.** Two triangulations T and T' are combinatorially equivalent if there exists a labeling which associates to each point $\mathbf{x_i}$ in T a point $\mathbf{x_i'}$ in T' so that the connectivity between $\mathbf{x_i}$'s induced by T matches the connectivity between the $\mathbf{x_i'}$'s induced by T'.

Definition 5 (Combinatorially Regular Triangulations (CRT))**.** A triangulation T of a d-dimensional point set X is called a **combinatorially regular triangulation** if there exists a d-dimensional point set X' admitting a regular triangulation T' such that T and T' are combinatorially equivalent.

Remark: these CRT triangulations have been introduced in [16] under the name of *weakly* regular triangulations, since a displacement of their vertices suffices to make them regular. Figure 3.2 (after [16]) shows an example of a combinatorially regular triangulation which is not, itself, regular.
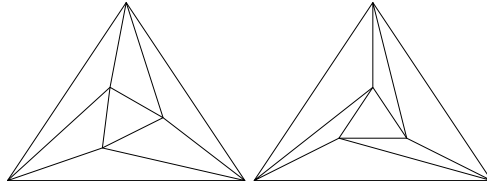
Figure 3.2: A regular triangulation (left), once deformed (right), becomes a combinatorially regular triangulation which is *not*, itself, regular.

Existence of PDTs in 2D. The 2D case is rather simple, due to this result mainly based on a classical theorem of Steinitz [27](see also [16] and [20]):

Proposition 6. *Any 2-dimensional triangulation is combinatorially regular.*

Therefore, every 2D triangulation T can be part of a PDT pair (T, K). However in higher dimensions (three and above), as we showed in [20], the situation is rather different:

Proposition 7. *For $d \geq 3$, there exist d-dimensional triangulations that do not admit any compatible dual complex.*

This is equivalent to the fact that there exist triangulations which are not combinatorially regular. The simplest non-combinatorially regular examples are the Brucker sphere and the Barnette sphere [9].

3.2. **PDT=CRT.** We claim that combinatorially regular triangulations are *the only ones that admit compatible dual complexes.* The proof revolves around a theorem due to Aurenhammer:

Every simple cell complex in $\mathbb{R}^d$, $d \geq 3$, is dual to a regular triangulation.

This theorem was proved in [2] through an iterative construction which is valid in any dimension $d \geq 3$. In [20], we used this theorem to prove the following theorem which surprisingly implies that in higher dimensions there are triangulations that do *not* admit a dual complex:

Theorem 8 (PDT Characterization)**.** *A d-dimensional triangulation T admits a compatible (not necessarily orthogonal) dual complex if and only if T is combinatorially regular.*

4. PARAMETERIZING PRIMAL-DUAL TRIANGULATIONS

We have established that primal-dual triangulations cover all dual complexes in $d \geq 3$; they also cover all $2D$ triangulations, but only triangulations which admit a dual in $d \geq 3$. We now focus on parameterizing the whole space of primal-dual triangulations with n points in $\mathbb{R}^d$ by simply adding parameters at the points. We then explore a geometric interpretation of this intrinsic parameterization as well as its properties.

The proof of Theorem 8 leads us very naturally to a parameterization of all the triangulations that admit a compatible dual complex:

Definition 9. A **parameterized primal-dual triangulation** is a primal-dual triangulation parameterized by a set of triplets $(\mathbf{x_i}, w_i, \mathbf{v_i})$, where $\mathbf{x_i}$ is the *position* in $\mathbb{R}^d$ of the ith node, w_i is a real number called the *weight* of x_i, and $\mathbf{v_i}$ is d-dimensional vector called the *displacement vector* of $\mathbf{x_i}$. The triangulation associated with the triplets $(\mathbf{x_i}, w_i, \mathbf{v_i})$ is defined such that its dual complex K is the *power diagram of weighted points* $(\mathbf{p_i}, w_i)$, where $\mathbf{p_i} = \mathbf{x_i} + \mathbf{v_i}$, see Fig 3.1.

The dual complex K can be seen as the generalized Voronoi diagram of the x_i's for the distance $d(\mathbf{x}, \mathbf{x_i}) = \|\mathbf{x} - \mathbf{x_i} - \mathbf{v_i}\|^2 - w_i$. When the vectors $\mathbf{v_i}$ are all null, the parameterized primal-dual triangulation T is regular, thus perpendicular to its dual K, and the pair (T, K) forms an orthogonal primal-dual triangulation. This proves that weighted Delaunay triangulations are sufficient to parameterize the set of all orthogonal primal-dual triangulations of a simply connected domain (see also [12]). The displacement vectors extend the type of triangulations and duals we can parameterize.

Characterizing the classes of *equivalent* triplets parameterizing the same PDT, and completed with constraints for the parameters in order to avoid redundancy between equivalent triplets, we find an efficient parameterization for the space of primal-dual triangulations (see [20] for details):

Theorem 10 (PDT Parametrization)**.** *There is a bijection between all primal-dual triangulations in $\mathbb{R}^d$ and sets of triplets $(\mathbf{x_i}, w_i, \mathbf{v_i})$, $1 \leq i \leq n$, where $\mathbf{x_i}, \mathbf{v_i} \in \mathbb{R}^d$, $w_i \in \mathbb{R}$ with $\sum_i w_i = 0$, $\sum_i \mathbf{v_i} = 0$, and $\sum_i \|\mathbf{x_i} + \mathbf{v_i}\|^2 = \sum_i \|\mathbf{x_i}\|^2$.*

Remark: using this parametrization, the particular case of Delaunay / Voronoi PDT of a set of points $\{\mathbf{x_i}\}_{i=1..n}$ is naturally parameterized by triplets $(\mathbf{x_i}, 0, 0)$. Note also that the condition $\sum_i w_i = 0$ may be replaced by $\min_i w_i = 0$, by simply subtracting the constant $\min_i w_i$ from all the weights of triplets (the PDT depending on weight differences will remain unchanged). This new condition implies that all the weights are positive which may be useful in some applications.

5. APPLICATIONS IN MESH OPTIMIZATION

In the previous section, we derived a natural parameterization of all non-orthogonal primal-dual structures of simply connected domains in $\mathbb{R}^d$. Besides the theoretical interest of these new primal-dual structures, we anticipate numerous applications. We believe that our results can benefit mesh optimization algorithms as we provide a particularly convenient way to explore a large space of primal-dual structures. We have already provided a first step in this direction by designing pairs of primal-dual structures that optimize accuracy bounds on differential operators using our parameterization [21], thus extending variational approaches designed to improve *either* primal (Optimal Delaunay Triangulations [28]) *or* dual (Centroidal Voronoi Tesselations) structures. In that work, we introduce Hodge-optimized triangulations (HOT), a family of well-shaped primal-dual pairs of complexes designed for fast and accurate computations in computer graphics. Other existing work most commonly employs barycentric or circumcentric duals: while barycentric duals guarantee that the dual of each simplex lies within the simplex, circumcentric duals are often preferred due to the induced orthogonality between primal and dual complexes. We instead promote the use of

weighted duals ("power diagrams"). They allow much greater flexibility in the location of dual vertices while keeping primal-dual orthogonality, thus providing an invaluable extension to the usual choices of dual by only adding one additional scalar per primal vertex. Furthermore, we introduce a family of functionals on pairs of complexes that we derive from bounds on the errors induced by diagonal Hodge stars, commonly used in discrete computations. The minimizers of these functionals, called HOT meshes, are shown to be generalizations of Centroidal Voronoi Tesselations and Optimal Delaunay Triangulations, and to provide increased accuracy and flexibility for a variety of computational purposes. This approach is detailed in the following sections.

6. Hodge Optimized Triangulations

To demonstrate the advantages of using regular/power triangulations, we focus on a particularly relevant type of functionals measuring primal and dual properties. Recall that for an arbitrary primal element σ, the diagonal approximation of the Hodge star $\star$ [4] of a continuous differential form α assumes

$$(6.1) \qquad \int_{*\sigma} \star\,\alpha \equiv \frac{|*\sigma|}{|\sigma|} \int_{\sigma} \alpha,$$

where $|.|$ denotes the Lebesgue measure (length, area, volume) of a simplex or cell. In other words, the discrete k^{th} Hodge star is encoded as a diagonal matrix $\star^k$ with $\forall i, \quad (\star^k)_{ii} := \frac{|*\sigma_i^k|}{|\sigma_i^k|}$, where σ_i^k (resp., $*\sigma_i^k$) is the ith k-simplex (resp., k-cell) of the primal-dual triangulation $\mathcal{M} = (\mathcal{T}, \mathcal{D})$; the discrete Hodge star of a discrete primal k-form ω^k is then computed as $\star^k \omega^k$, and the extension to dual discrete forms (with, this time, $(\star^k)^{-1}$) is trivial (for further details see, e.g., [6]). We will link approximation error of diagonal Hodge stars to optimal transport.

6.1. **Basics of Optimal Transport.** The optimal transport problem seeks to determine the optimal way to move a pile of dirt M to a hole N of the same volume, where "optimal" means that the integral of the distances by which the dirt is moved is minimal. The notion of "distance" (i.e., cost of transport) may vary based on context. A common distance function defined between probability distributions in $\mathbb{R}^d$ with bounded support is the q-Wasserstein metric, defined as

$$W_q(\mu, \nu) = \left(\inf_{\pi \in \mathcal{P}(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|^q \; \mathrm{d}\pi(x,y) \right)^{1/q}.$$

Let us recall here the Kantorovich-Rubinstein theorem, stating that for two distributions μ and ν with bounded support, the 1-Wasserstein distance between μ and ν can be rewritten as:

$$(6.2) \qquad W_1(\mu, \nu) = \sup_{\substack{\text{continuous } \varphi:S\to\mathbb{R} \\ \text{Lips}(\varphi)\leq\lambda}} \frac{1}{\lambda} \int_S \varphi(x)\,\mathrm{d}(\mu - \nu),$$

where $\text{Lips}(\varphi)$ represents the Lipschitz constant of function φ. This expression will be useful shortly to link optimal transport and approximation error of diagonal Hodge stars.

6.2. **Deriving Tight Bounds through Optimal Transport.** While computationally convenient, diagonal Hodge stars are obviously not very accurate: they are generally only exact for constant forms. We can quantify the induced inaccuracy of $\star^k$ by defining the *error density* e_i on the dual of the simplex σ_i as the average difference between the discrete approximation and the real Hodge star value:

$$e_i := \frac{1}{|*\sigma_i|} \left| \frac{|*\sigma_i|}{|\sigma_i|} \int_{\sigma_i} \omega - \int_{*\sigma_i} \star\omega \right| = \left| \frac{1}{|\sigma_i|} \int_{\sigma_i} \omega - \frac{1}{|*\sigma_i|} \int_{*\sigma_i} \star\omega \right|$$

$$= \left| \int_{\sigma_i} f(x)\frac{d\sigma_i}{|\sigma_i|} - \int_{*\sigma_i} f(x)\frac{d*\sigma_i}{|*\sigma_i|} \right| = \left| \int_{\sigma_i \cup *\sigma_i} f(x) \left[\frac{d\sigma_i}{|\sigma_i|} - \frac{d*\sigma_i}{|*\sigma_i|} \right] \right|,$$

where $f(x)$ is the component of ω on $d\sigma_i$. We deduce, using Eq.(6.2), that the *tightest bound* one can find on the Hodge star error density per simplex for an arbitrary λ-Lipschitz form is simply λ times the minimum cost over all transport plans between σ_i (seen as a uniform distribution over the mesh element that integrates to one) and $*\sigma_i$ (also seen as a uniform distribution integrating to one); i.e., with a slight abuse of notation,

$$(6.3) \qquad e_i \leq \lambda\, W_1(\sigma_i, *\sigma_i).$$

This formally establishes a link between Hodge star accuracy and optimal transport. Note that we only required ω to be Lipschitz continuous, a reasonable assumption in most graphics applications.

6.3. **Error Functionals on Meshes.** From these local error densities, we can assemble a total error by taking the $L_{p \geq 1}$ integral norm of the error over the mesh area by integrating the pth power of the error density over each convex hull[1] of σ_i and its dual $*\sigma_i$. This directly yields:

$$E_p(\mathcal{M}, \star^k) = \left(\sum_{\sigma_i \in \Sigma^k} \int_{\text{CH}(\sigma_i \cup *\sigma_i)} e_i{}^p \right)^{\frac{1}{p}} = \left(\sum_{\sigma_i \in \Sigma^k} \frac{|\sigma_i| |*\sigma_i|}{\binom{k}{d}} e_i{}^p \right)^{\frac{1}{p}},$$

since the convex hull $\text{CH}(\sigma_i \cup *\sigma_i)$ is, up to a dimension factor, simply the product of the primal and dual volumes due to our primal/dual orthogonality assumption of mesh $\mathcal{M}$. Note that these convex hulls, coined "support volumes in" [15] and "diamonds" in [14, 6], tile the whole primal mesh, thus providing a proper volume integral.

From Eq. (6.3), we conclude that a tight bound for the pth power of the total error is expressed as:

$$(6.4) \qquad E_p(\mathcal{M}, \star^k)^p \leq \frac{\lambda}{\binom{k}{d}} \sum_{\sigma_i \in \Sigma^k} |*\sigma_i| |\sigma_i| \, W_1(\sigma_i, *\sigma_i)^p.$$

(Notice that $E_\infty(\mathcal{M}, \star^k)$ is thus, up to the Lipschitz constant, bounded by the maximum of the minimum W_1 distance between primal and dual elements of the mesh as expected.) For notational convenience, we will denote by $\star^k\text{-}\text{HOT}_{p,1}(\mathcal{M})$ the bound (with Lipschitz and dimension constants removed) obtained in Eq. (6.4); more generally, we will define

$$\boxed{\star^k\text{-}\text{HOT}_{p,q}(\mathcal{M}) \equiv \sum_{\sigma_i \in \Sigma^k} |*\sigma_i| |\sigma_i| \, W_q(\sigma_i, *\sigma_i)^p}$$

as relevant *functionals* (or energies) to construct meshes, as minimizing them will control the quality of the discrete Hodge stars. Let us note that most of our HOT energies are evaluated by splitting simplices/cells into canonical subsimplices for which closed-forms integral expressions $W(\mathbf{p}, T)$ of simplex-T-to-point-$\mathbf{p}$ transport are easily found.

Discussion: Our HOT energies are archetypical, general-purpose examples of mesh quality measures imposed on both primal *and* dual meshes, but they are by no means unique: from the local error densities e_i, other energies can be formulated to target more specific errors occurring in mesh computations. Note also that the use of a 1-Wasserstein distance is notably less attractive numerically than a 2-Wasserstein distance. Fortunately, we can also provide a bound of the Hodge star error which, while less tight than the previously derived $\text{HOT}_{p,1}$, will be particularly convenient to deal with computationally:

$$E_2(\mathcal{M}, \star^k)^2 \leq \sum_{\sigma_i \in \Sigma^k} |*\sigma_i| |\sigma_i| \, W_2(\sigma_i, *\sigma_i)^2 \equiv \star^k\text{-}\text{HOT}_{2,2}(\mathcal{M}).$$

The reader may have noticed that the functional $\star^0\text{-}\text{HOT}_{2,2}(\mathcal{M})$ is, in the case of equal weights, the well-known Centroidal Voronoi Tesselation (CVT) energy $\left(\sum_i \int_{V_i} \| \mathbf{x} - \mathbf{x}_i \|^2 \, dV \right)$ for which several minimization techniques, from Lloyd iterations [7] to quasi-Newton methods [18], have been developed. L_p variants (i.e., $\star^0\text{-}\text{HOT}_{2p,2}(\mathcal{M})$ for $p \geq 2$) were also explored recently [17]. However, these energies only correspond to $\star^0$, and are not as tight as $\text{HOT}_{1,p}$. Our HOT energies can thus be seen as a direct generalization of the CVT-like functionals. Note finally that the Optimal Delaunay Triangulation (ODT) energy used in [1] can also be seen as a variant of $\star^d\text{-}\text{HOT}_{2,2}(\mathcal{M})$ in $\mathbb{R}^d$ for which the dual mesh is restricted to be "barycentric"; alas, the resulting mesh will not necessarily lead to an orthogonal primal-dual triangulation—even if the resulting simplices were proven to be very close to isotropic.

---

[1]The reader may notice that when σ_i and $*\sigma_i$ do *not* intersect, the integration domain is no longer a convex set, but a signed union of subsimplices. For simplicity, we will still use the term "convex hull".

6.4. **General Minimization Procedure.** Given that both (continuous) vertex positions and (discrete) mesh connectivity need to be optimized, the task of finding HOT meshes is seemingly intractable. Thankfully, regular triangulations provide a good *parameterization* of the type of primal-dual meshes we wish to explore: one can simply *continuously optimize* both positions and associated weights to find a HOT mesh. However, HOT energies are not convex in general, and a common downfall of non-convex optimization is its propensity to settle into local minima. In our case, this can be nicely alleviated by starting the optimization from a mesh relatively near the optimal solution, i.e., for which the vertices are well spread out. We thus run, after initializing the domain with uniformly sampled vertices over the domain, a few iterations of CVT [7] or ODT [1] to quickly disperse the vertices and get mesh elements roughly similar in size. Once a good initial mesh has been found, we perform a gradient descent, or alternatively, an L-BFGS algorithm [23]. A linear search is performed to adapt the step size along the gradient or the quasi-Newton direction. This common minimization procedure works quite well without requiring anything else but an evaluation of our HOT energies and their gradients. Note that since the positions $\mathbf{x}_i$ and the weights w_i have very different scales, we proceed by alternatively minimizing our HOT energies with respect to vertex positions and weights.

7. Results

HOT meshes can be beneficial in a number of contexts in modeling of surfaces and volumes, as well as in simulation. A particularly common linear operator in mesh processing is the Laplacian Δ, be it in the plane or on a discrete surface. As its Discrete Exterior Calculus (DEC) expression for 0-forms is $\Delta = d_0^t \star^1 d_0$, our HOT energies for $\star^1$ should be particularly adapted to its accurate computation: the d_0 operator being exact, the only loss of accuracy rises from this particular Hodge star. A $\star^1$-HOT$_{2,2}$ mesh indeed results, on a discretization of a simple test domain with 200 vertices, in a 5% reduction of the condition number of the Laplacian matrix with Dirichlet boundary conditions compared to a CVT mesh. The result is much more dramatic for the Laplacian of *dual* 0-forms, where the condition number drops from 254 to 90 on the same example.

To some extent, our approach can even help to deal with situations where the primal triangulation is given and cannot safely be altered: for instance, moving vertices and/or changing the connectivity of a triangle mesh in $\mathbb{R}^3$ is potentially harmful, as it affects the surface shape. Still, the ability to optimize weights to drive the selection of the dual mesh is very useful. We can easily optimize primal-dual triangulations (meshes) by minimizing a functional (energy) with respect to weights. The connectivity is kept intact, regardless of the weights—only the position and shape of the compatible dual is optimized. Our 2D and 3D experiments [21] show that only optimizing the weights is particularly simple and beneficial on a number of meshes. Fig. 7.1 depicts a triangle mesh of a hand and its intrinsic dual before and after weight optimization, showing a drastic reduction in the number of negative dual edges—thus providing a practical alternative to the use of intrinsic Delaunay meshes advocated in [10].

As another illustrative example, Fig. 8.1 shows that even an optimized Delaunay triangulation (ODT mesh, 195K tets, 36K vertices)) with exceptionally high-quality tetrahedra [28] can be made significantly better centered (i.e., with dual vertices closer to the inside of their associated primal simplex) using a simple weight optimization. Note also that in this example the number of tetrahedra with a dual vertex outside of the primal tet dropped from 17041 on the ODT mesh to 5489 on the optimized mesh—a two third reduction of *outcentered* tetrahedra.

8. Conclusion

We introduced the notion of compatible dual complex for a given triangulation in $\mathbb{R}^d$, and discussed the conditions under which an arbitrary triangulation of a simply connected domain admits a compatible, possibly non-orthogonal dual complex. Note that our only requirement for the dual is that it is made out of *convex polytopes*, thus reducing the space of possible primal-dual pairs to a computationally-convenient subset for which basis functions and positive barycentric coordinates are easily defined. We also pointed out a link to a previously-introduced notion of weakly regular triangulation by Lee in the nineties, and that there are triangulations that do *not* admit a dual complex. In addition, the parameterization we derived for all non-orthogonal primal-dual structures, provides a particularly convenient way to explore a large space of triangulations
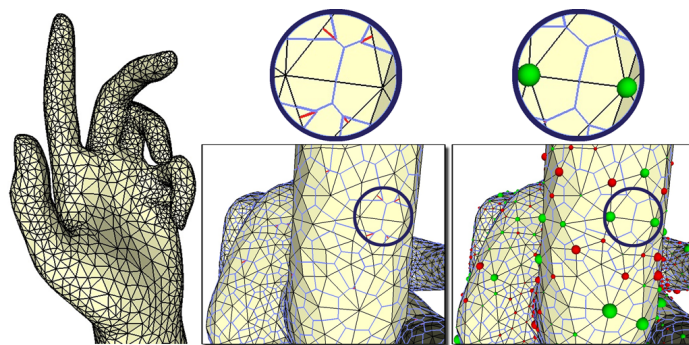
Figure 7.1: **Improving Dual Structure of a Surface Mesh**: For a given triangular mesh (left) there are several triangles whose circumcenter is far outside the triangle (center, lines drawn in red). By optimizing only the weights the new dual vertices are better placed inside the unchanged triangles (right) while keeping primal/dual orthogonality.

for which we anticipate numerous applications. As we have shown in the last sections of this note, our results can in particular benefit mesh optimization algorithms. In addition to applications in mesh optimization, modeling (as in computational biology) that uses *convex space tilings* could directly use our parameterization of PDTs. Clustering techniques based on k-means may also benefit from parameterizing clusters by more than just centers, as weights and vectors add more flexibility to the segmentation of input data.
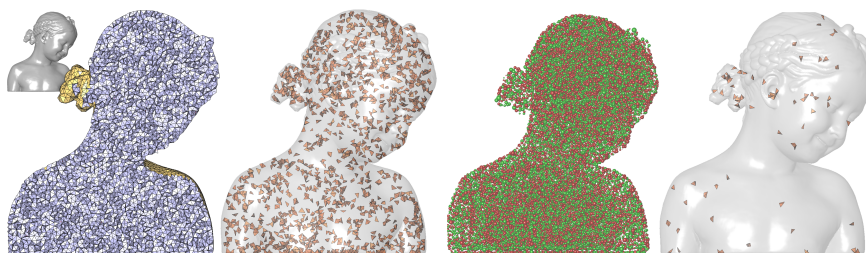
Figure 8.1: **Improving Dual Structure of 3D Meshes**: A the dual of a high-quality ODT mesh of the Bimba con Nastrino (a) can be optimized in terms of Hodge star operator's accuracy; by improving minimal dual edge length and self-centeredness. (c) Weights are displayed according to sign (red/green) and magnitude. When we single out the tetrahedra with a distance between weighted circumcenter and barycenter greater than 0.5% of the bounding box, one can see the optimized mesh (d) is significantly better than the original ODT (b), even if the primal triangulations are exactly matching.

References

[1] Pierre Alliez, David Cohen-Steiner, Mariette Yvinec, and Mathieu Desbrun. Variational tetrahedral meshing. *ACM Trans. on Graphics (SIGGRAPH)*, 24(3):617–625, July 2005.

[2] F. Aurenhammer. A criterion for the affine equivalence of cell complexes in $\mathbb{R}^d$ and convex polyhedra in $\mathbb{R}^{d+1}$. *Discrete and Computational Geometry*, 2(1):49–64, 1987.

[3] B. R. Baligaa and S. V. Patankarb. A Control Volume Finite-Element Method For Two-Dimensional Fluid Flow And Heat Transfer. *Numerical Heat Transfer*, 6:245–261, 1983.

[4] Alain Bossavit. *Computational Electromagnetism*. Academic Press, Boston, 1998.

[5] Fernando De Goes, Pierre Alliez, Houman Owhadi, Mathieu Desbrun, et al. On the equilibrium of simplicial masonry structures. *ACM Transactions on Graphics*, 32(4), 2013.

[6] Mathieu Desbrun, Eva Kanso, and Yiying Tong. Discrete differential forms for computational modeling. In A. Bobenko and P. Schröder, editors, *Discrete Differential Geometry*. Springer, 2007.

[7] Qiang Du, Vance Faber, and Max Gunzburger. Centroidal voronoi tessellations: Applications and algorithms. *SIAM Rev.*, 41:637–676, December 1999.

[8] H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer-Verlag, 1987.

[9] G. Ewald. *Combinatorial convexity and algebraic geometry*. Springer Verlag, 1996.

[10] Matthew Fisher, Boris Springborn, Alexander I. Bobenko, and Peter Schröder. An algorithm for the construction of intrinsic delaunay triangulations with applications to digital geometry processing. In *ACM SIGGRAPH Courses*, pages 69–74, 2006.

[11] I.M. Gelfand, M.M. Kapranov, and A.V. Zelevinsky. *Discriminants, resultants, and multidimensional determinants*. Springer, 1994.

[12] D. Glickenstein. Geometric triangulations and discrete Laplacians on manifolds. *Arxiv preprint math/0508188*, 2005.

[13] Leo J. Grady and Jonathan R. Polimeni. *Discrete Calculus: Applied Analysis on Graphs for Computational Science*. Springer, 2010.

[14] P. Hauret, E. Kuhl, and M. Ortiz. Diamond Elements: a finite element/discrete-mechanics approximation scheme with guaranteed optimal convergence in incompressible elasticity. *Int. J. Numer. Meth. Engng.*, 72(3):253–294, 2007.

[15] A. N. Hirani. *Discrete Exterior Calculus*. PhD thesis, Caltech, May 2003.

[16] C.W. Lee. Regular triangulations of convex polytopes. *Applied Geometry and Discrete Mathematics–The Victor Klee Festschrift (P. Gritzmann, B. Sturmfels, eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Amer. Math. Soc*, 4:443–456, 1991.

[17] Bruno Lévy and Yang Liu. L_p Centroidal Voronoi Tesselation and its Applications. *ACM Trans. on Graph.*, 29(4), 2010.

[18] Yang Liu, Wenping Wang, Bruno Lévy, Feng Sun, DongMing Yan, Lin Lu, and Chenglei Yang. On Centroidal Voronoi Tessellation - energy smoothness and fast computation. *ACM Trans. on Graph.*, 28(4), 2009.

[19] S. F. McCormick. *Multilevel Adaptive Methods for Partial Differential Equations*. SIAM, 1989.

[20] Pooran Memari, Patrick Mullen, and Mathieu Desbrun. Parametrization of generalized primal-dual triangulations. In *Proceedings of the 20th International Meshing Roundtable*, pages 237–253. Springer, 2012.

[21] Patrick Mullen, Pooran Memari, Fernando de Goes, and Mathieu Desbrun. Hot: Hodge-optimized triangulations. *ACM Transactions on Graphics (TOG)*, 30(4):103, 2011.

[22] O.R. Musin. Properties of the Delaunay triangulation. In *Symposium on Computational Geometry*, page 426, 1997.

[23] J. Nocedal and S. J. Wright. *Numerical optimization*. Springer Verlag, 1999.

[24] A. Paluszny, S. Matthäi, and M. Hohmeyer. Hybrid finite element finite volume discretization of complex geologic structures and a new simulation workflow demonstrated on fractured rocks. *Geofluids*, 7:186–208, 2007.

[25] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, 1985.

[26] VT Rajan. Optimality of the Delaunay triangulation in $\mathbb{R}^d$. *Discrete and Computational Geometry*, 12(1):189–202, 1994.

[27] E. Steinitz. Polyeder und raumeinteilungen. *Encyclopädie der mathematischen Wissenschaften*, 3(9):1–139, 1922.

[28] Jane Tournois, Camille Wormser, Pierre Alliez, and Mathieu Desbrun. Interleaving delaunay refinement and optimization for practical isotropic tetrahedron mesh generation. *ACM Trans. Graph.*, 28:75:1–75:9, July 2009.

[29] G.M. Ziegler. *Lectures on polytopes*. Springer, 1995.

CNRS-LTCI, Télécom-ParisTech • memari@telecom-paristech.fr

Variational properties of the discrete Hilbert-Einstein functional

Ivan IZMESTIEV

1. INTRODUCING THE FUNCTIONAL

1.1. Smooth case. Let M be a smooth compact manifold without boundary. In Riemannian geometry, the *Hilbert-Einstein functional* is a function on the space Met_M of Riemannian metrics on M which associates to a metric g the integral of half its scalar curvature:

$$S \colon Met_M \to \mathbb{R}, \quad S(g) = \frac{1}{2} \int_M \mathrm{scal}_g \, \mathrm{dvol}_g$$

If $\dim M = 2$, then we have $\mathrm{scal}_g = 2K_g$, where K_g is the Gauss curvature. Hence by the Gauss-Bonnet theorem

$$S(g) = 2\pi\chi(M)$$

is independent of the metric g.

Starting from $\dim M = 3$, the functional S becomes more interesting. Denote

$$S'_h = \frac{d}{dt}\bigg|_{t=0} S(g + th),$$

where h is the field of symmetric bilinear forms on M.

Theorem 1.1. *The first variation of S is given by the formula*

$$S'_h = \frac{1}{2} \int_M \left\langle \frac{\mathrm{scal}_g}{2} g - \mathrm{Ric}_g, h \right\rangle \mathrm{dvol}_g$$

Corollary 1.2. *Let $\dim M \geq 3$.*
  a) *A metric $g \in Met_M$ is a critical point of S if and only if g is Ricci-flat, i. e. $\mathrm{Ric}_g = 0$.*
  b) *Critical points of the restriction of S to the space Met^1_M of metrics of unit total volume are Einstein metrics, i. e. metrics with $\mathrm{Ric}_g = \lambda g$.*

If $\dim M = 3$, then Einstein metrics are metrics of constant sectional curvature (Euclidean, hyperbolic or spherical).

See [4, Chapter 4C] for details.

1.2. Discrete case. Let M be a compact 3-manifold without boundary. Fix a triangulation (i. e. simplicial face-to-face decomposition) T of M and pick a map

$$\ell \colon \mathcal{E}(T) \to (0, +\infty), \quad e \mapsto \ell_e$$

assigning to every edge e of T a length ℓ_e. Consider only those ℓ for which every tetrahedron of T can be realized as a Euclidean tetrahedron with the edge lengths ℓ. (This set is non-empty, since $\ell_e = 1$ for all e will do.)

The map ℓ introduces a Euclidean metric on each tetrahedron of T, and a *Euclidean cone-metric* on M. Note that different pairs (T, ℓ) can define the same metric; for example, we may subdivide the triangulation T and define lengths of new edges appropriately.

The *Hilbert-Einstein functional* on the space of Euclidean cone-metrics is

$$S(T, \ell) = \sum_{e \in \mathcal{E}(T)} \ell_e (2\pi - \omega_e),$$

where ω_e is the total angle around e, see Figure 1.1. Clearly, the value of S depends only on the metric, not on the choice of the representative (T, ℓ).
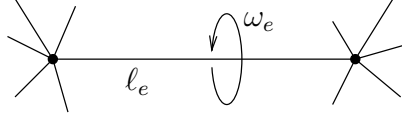

Figure 1.1: Lengths and angles in a 3-dimensional cone-manifold.

Remark 1.3. If $\dim M = n$, then Euclidean cone-metrics on M have cone singularities around codimension 2 faces of T, and one puts

$$S(T, \ell) = c_n \sum_{\dim F = n-2} \mathrm{vol}_{n-2}(F)(2\pi - \omega_F)$$

for some constant c_n. Cheeger, Müller, and Schrader [8] have shown that the discrete Hilbert-Einstein functional converges to the smooth one if a sequence $(T^{(n)}, \ell^{(n)})$ of Euclidean cone-metrics converges to a Riemannian metric g (with respect to the Lipschitz distance between metric spaces) so that all simplices in $(T^{(n)}, \ell^{(n)})$ stay sufficiently fat.

It is an open problem to what most general class of metric spaces (including Riemannian manifolds and Euclidean cone-manifolds) the Hilbert-Einstein functional, and more generally, all total Lipschitz-Killing curvatures can be extended.

The Hilbert-Einstein functional can also be defined for hyperbolic and spherical cone-metrics. In this case an additional volume term appears, see [14, Sections 4.2-4.4].

1.3. **Critical points in the discrete case.** Call the quantity $\kappa_e := 2\pi - \omega_e$ the *curvature* of a Euclidean cone-metric at the edge e. Then we have $S(T, \ell) = \sum_e \ell_e \kappa_e$.

Theorem 1.4. *We have $\frac{\partial S}{\partial \ell_e} = \kappa_e$*

This is equivalent to the identity $\sum_e \ell_e d\kappa_e = 0$, which follows by adding up the Schläfli formula for all tetrahedra of T. An independent proof was given by the physicist Tullio Regge who introduced the discrete Hilbert-Einstein functional in [18]. In particular, Regge's argument provides an elementary proof of the Schläfli formula.

Corollary 1.5. *Critical points of the discrete Hilbert-Einstein functional represent flat metrics.*

Similarly, critical points of the functional on the space of hyperbolic cone-metrics (see end of Section 1.2) correspond to hyperbolic metrics without cone singularities.

Corollary 1.5 has two applications:

- Construct a metric of constant curvature by finding a critical point of S .
- Prove rigidity of a space-form by showing non-degeneracy of the corresponding critical point of S.

It is surely tempting to try to reprove hyperbolization theorem by showing the existence of critical points of S under suitable topological assumptions on M. Two main difficulties arise here. One is that the functional is neither convex nor concave, which makes existence of a critical point difficult to prove. The other is the choice of a triangulation T, since we cannot know in advance the combinatorial type of a geodesic triangulation. One possible solution is to start with an arbitrary triangulation and change its combinatorial type while deforming the metric. This is what was done in our proof of the Alexandrov theorem (Section 3) which can be viewed as a simple case of geometrization with boundary conditions.

In the smooth case, Blaschke and Herglotz [5] suggested to use the variational property of the smooth Hilbert-Einstein functional for solving Weyl's problem. Yamabe's work [22] resulted from an attempt to solve Poincaré's conjecture using the same variational principle. Most recently, a geometrization program developing Yamabe's ideas was proposed by M. Anderson [2, 3].

The second of the above points, the infinitesimal rigidity, is more easily tractable. Variational properties of the Hilbert-Einstein functional form the basis of Koiso's proof of the infinitesimal rigidity of Einstein manifolds under certain assumptions on the eigenvalues of the curvature tensor, [16]. We used similar ideas in a new proof of the infinitesimal rigidity of convex polyhedra (Section 2) and of a class of non-convex polyhedra (Section 4).

2. Infinitesimal rigidity of convex polyhedra

2.1. **The boundary term of the Hilbert-Einstein functional.** If the compact manifold M has a non-empty boundary, then the Hilbert-Einstein functional needs a boundary term, in order to remain differentiable. In the smooth case, this is

$$S(g) = \frac{1}{2} \int_M \mathrm{scal}_g \, \mathrm{dvol}_g + \int_{\partial M} H \, \mathrm{dvol}_g^\partial,$$

where H is the trace of the second fundamental form II. The variational formula becomes

$$S'_h = \frac{1}{2} \int_M \left\langle \frac{\mathrm{scal}_g}{2} g - \mathrm{Ric}_g, h \right\rangle \mathrm{dvol}_g + \frac{1}{2} \int_{\partial M} \langle Hg - II, h \rangle \, \mathrm{dvol}_g^\partial$$

In the discrete case we have

$$S(T, \ell) = \sum_{e \in \mathcal{E}_i(T)} \ell_e (2\pi - \omega_e) + \sum_{e \in \mathcal{E}_\partial(T)} \ell_e (\pi - \theta_e),$$

where $\mathcal{E}_i(T)$ and $\mathcal{E}_\partial(T)$ are the sets of interior and boundary edges of T, respectively, and θ_e is the dihedral angle at the boundary edge e. The variational formula is obtained again by adding up the Schläfli formulas for individual simplices:

$$\frac{\partial S}{\partial \ell_e} = \begin{cases} 2\pi - \omega_e, & \text{if } e \in \mathcal{E}_i(T) \\ \pi - \theta_e, & \text{if } e \in \mathcal{E}_\partial(T) \end{cases}$$

Remark 2.1. If $M \subset \mathbb{R}^3$ is a convex body, then both of the above boundary terms appear as the coefficients at the t^2 term in the Steiner formula for M. Another common interpretation of both is 4π times the mean width (average length of projections to lines) of M. Check ball and cube.

If we keep the metric on the boundary fixed (that is $h(X, Y) = 0$ for $X, Y \in T\partial M$ in the smooth case, and $\ell_e = \mathrm{const}$ for $e \in \mathcal{E}_\partial(T)$ in the discrete case), then the critical points of the functionals are metrics that are flat inside M and restrict to the given metric on the boundary.

2.2. **A proof of the infinitesimal rigidity of a convex polyhedron.** Let $P \subset \mathbb{R}^3$ be a compact convex polyhedron. For simplicity, assume that all faces of P are triangles. An *infinitesimal isometric deformation* of P is an assignment of a vector q_i to every vertex p_i such that

$$\langle p_i - p_j, q_i - q_j \rangle = 0 \quad \text{for every edge } p_i p_j$$

which is equivalent to $\frac{\partial}{\partial t}\big|_{t=0} \|p_i(t) - p_j(t)\| = 0$ with $p_i(t) = p_i + t q_i$. A polyhedron is called *infinitesimally rigid* if every its infinitesimal isometric deformation extends to an infinitesimal isometry of $\mathbb{R}^3$.

We will take another viewpoint: instead of deforming an embedded surface (the boundary of the polyhedron) we deform the metric inside the polyhedron itself. For this, choose a point a inside P and subdivide P into triangular pyramids with apex a and faces of P as bases. This results in a triangulation T of P. Denote by

$$r_i := \|a - p_i\|, \quad \ell_{ij} := \|p_i - p_j\|$$

the lengths of interior and boundary edges, respectively. We will change r_i while keeping ℓ_{ij} fixed and look what happens with the curvatures κ_i around the interior edges (at the beginning we have $\kappa_i = 0$). See Figure 2.1.
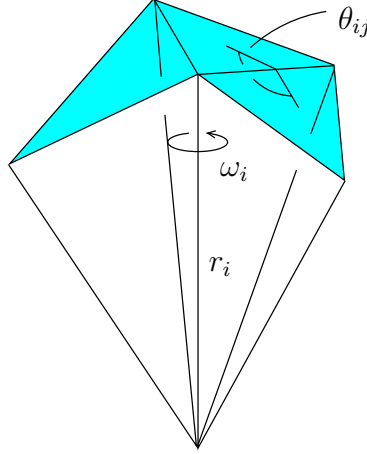
Figure 2.1: Lengths and angles in the triangulation T of the polyhedron P. Shaded triangles lie on the boundary; only a part of the triangulation is shown.

Definition 2.2. *A deformation* (s_i) *of the interior edge lengths* (r_i) *is called* curvature-preserving, *if all directional derivatives*

$$\frac{d\kappa_i}{ds} = \sum_j \frac{\partial \kappa_i}{\partial r_j} s_j$$

vanish. In other words, if

$$s \in \ker\left(\frac{\partial \kappa_i}{\partial r_j}\right) = \ker\left(\frac{\partial^2 S}{\partial r_i \partial r_j}\right)$$

Among curvature-preserving deformations there are trivial ones that result from a displacement of the point a inside P. It is easy to show that they form a 3-dimensional subspace. Also it is easy to see that the infinitesimal rigidity of P in the original sense is equivalent to the absence of non-trivial curvature-preserving deformations:

$$P \text{ is infinitesimally rigid } \Leftrightarrow \dim\ker\left(\frac{\partial^2 S}{\partial r_i \partial r_j}\right) = 3$$

The following theorem implies that convex polyhedra are infinitesimally rigid.

Theorem 2.3. *Let P be a compact convex polyhedron with triangular faces and triangulation T as described above. Then the second variation $\left(\frac{\partial^2 S}{\partial r_i \partial r_j}\right)$ of the discrete Hilbert-Einstein functional has the signature $(+, 0, 0, 0, -, \dots, -)$.*

The part about the rank of the second variation is proved in [14, Section 3]. The fact that the positive index is equal to 1 follows from the coincidence of the second variations of S and of the volume of polar dual [14, Section 4.1], and from the signature of the second variation of the volume, provided by the second Minkowski inequality for mixed volumes, [12, Appendix].

3. ALEXANDROV'S THEOREM

Alexandrov's theorem [1] states the existence and uniqueness of a compact convex polyhedron in $\mathbb{R}^3$ with a prescribed boundary metric. The intrinsic boundary metric is a Euclidean cone-metric (since the surface of a polyhedron can be glued from triangles) with singular points of positive curvature (vertices of the polyhedron). Note that the intrinsic metric does not detect the edges of a polyhedron.

Theorem 3.1 (A. D. Alexandrov, [1])**.** *Let g be a Euclidean cone-metric on the sphere with singular points of positive curvature. Then there exists a unique up to congruence compact convex polyhedron in $\mathbb{R}^3$ with g as the intrinsic metric on the boundary. (The polyhedron may also be a polygon, in which case instead of the intrinsic metric on the boundary two copies of the polygon glued along pairs of corresponding edges are taken.)*

154

In [6] a new proof of Alexandrov's theorem was given, similar in the spirit to the proof of the infinitesimal rigidity described in Section 2.2.

We start with a certain geodesic triangulation $\bar{T}(0)$ of the sphere equipped with metric g, with vertices at the singular points, and an assignment of a positive number $r_i(0)$ to every singular point p_i. This allows us to construct a Euclidean cone-manifold $P(\bar{T}(0), r(0))$ by gluing together triangular pyramids with radial edge lengths $r_i(0)$ and triangles of $\bar{T}_0$ as bases. Namely, we take the Delaunay triangulation of $(\mathbb{S}^2, g)$ as $\bar{T}(0)$, and put $r_i(0) = R$ for all i, with R sufficiently large. This ensures that pyramids exist and that the "warped polyhedron" $P(\bar{T}(0), r(0))$ is convex at the boundary (i. e. $\theta_{ij}(0) \le \pi$).

Then we proceed by deforming the lengths r_i, thus obtaining a continuous family of warped polyhedra $P(\bar{T}(t), r(t))$. The deformation is chosen so that

- $\kappa_i(t) = (1-t)\kappa_i(0)$, where $\kappa_i(t)$ is the curvature at the edge ap_i in $P(\bar{T}(t), r(t))$;
- the dihedral angles on the boundary remain $\le \pi$.

The second condition requires that at certain moments $t_1 < t_2 < \dots$ the triangulation $\bar{T}(t)$ must be changed. The triangulation is determined uniquely (up to "flat edges", those where the dihedral angle is π) since the second condition is equivalent to $\bar{T}(t)$ being the weighted Delaunay triangulation of $(\mathbb{S}^2, g)$ with weights r_i^2, see [6, Section 2.5].

The existence of a deformation satisfying the first condition follows from the non-degeneracy of the matrix $\left(\frac{\partial \kappa_i}{\partial r_j}\right)$ under certain assumptions [6, Theorem 3.11–Proposition 3.16]. In the limit as $t \to 1$ we have $\kappa_i \to 0$ for all i, thus $P(\bar{T}(1), r(1))$ is a compact convex polyhedron with a given metric on the boundary.

A corresponding numerical algorithm was implemented in a computer program by Stefan Sechelmann [19], see Figure 3.1.
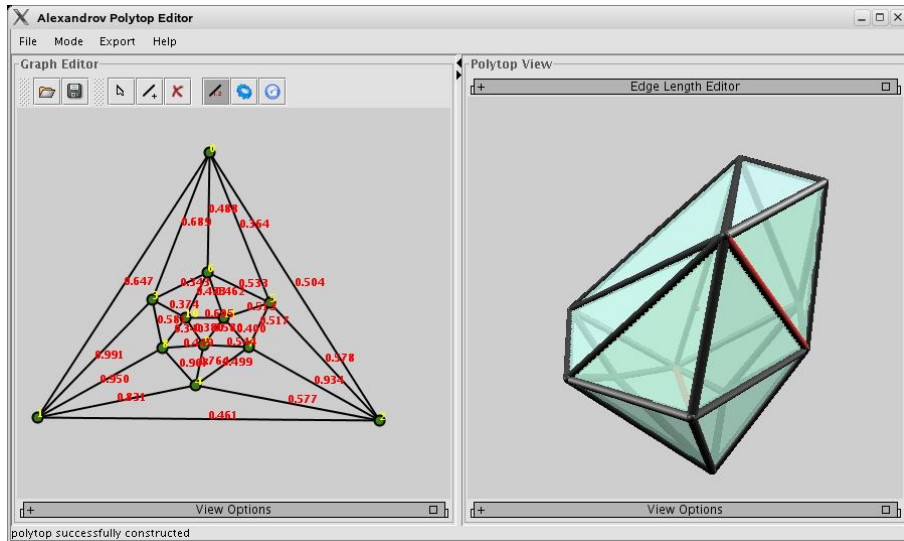
Figure 3.1: A screenshot of [19].

4. Infinitesimal rigidity of weakly convex (co)decomposable polyhedra

Infinitesimally flexible non-convex polyhedra exist, see Figure 4.1.

In [15] the infinitesimal rigidity was proved for a wide class of non-convex polyhedra.

Definition 4.1. *A non-convex polyhedron is called* weakly convex, *if its vertices lie in a convex position:* $\mathrm{Vert}(P) = \mathrm{Vert}(\mathrm{conv}\, P)$.

A weakly convex polyhedron P is called decomposable *if it can be triangulated without adding new vertices. It is called* decomposable and codecomposable *if there is a triangulation T of $\mathrm{conv}\, P$ such that* $\mathrm{Vert}(T) = \mathrm{Vert}(P)$ *and P is a subcomplex of T.*

155
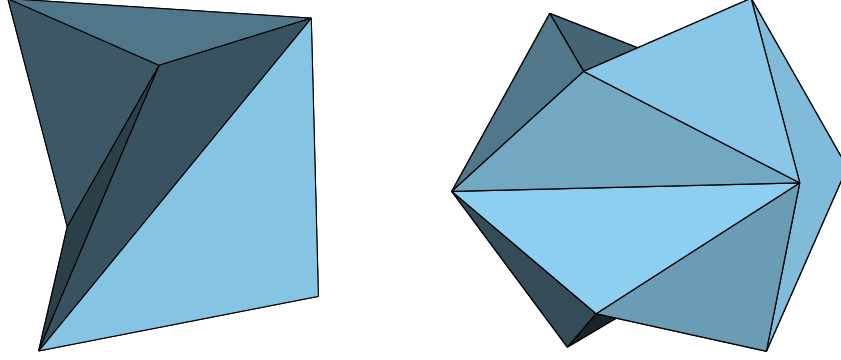
Figure 4.1: Schoenhardt's twisted octahedron and Jessen's orthogonal icosahedron are infinitesimally flexible.

Both polyhedra on Figure 4.1 are weakly convex but not decomposable.

Theorem 4.2 ([15], Theorem 1.7). *Weakly convex decomposable and codecomposable polyhedra are infinitesimally rigid.*

This theorem is a consequence of the following property of the Hilbert-Einstein functional.

Theorem 4.3 ([15], Theorem 1.17). *Let T be a triangulation of a convex polyhedron. Denote by i the number of vertices of T in the interior of the polyhedron and by b the number of vertices in the interiors of its faces. (The number of vertices on the edges is irrelevant.)*

Consider Euclidean cone-metrics inside the polyhedron arising from deformations of the interior edges of the triangulation. Then the matrix

$$(4.1) \qquad \left(\frac{\partial \kappa_e}{\partial \ell_f}\right) = \left(\frac{\partial^2 S}{\partial \ell_e \partial \ell_f}\right)$$

where ℓ_e, ℓ_f denote the lengths of interior edges, has corank $3i+b$ and exactly i positive eigenvalues.

Corollary 4.4. *Let T be a triangulation of a convex polyhedron that uses only vertices of this polyhedron. Then the matrix (4.1) is negative definite.*

Proof of Theorem 4.2. In the triangulation T of conv P, take the subcomplex $\bar{T}$ that triangulates P. The Hessian matrix of S for $\bar{T}$ is a principal minor of the Hessian of S for T. Since the latter is negative definite, so is the former. In particular, it is non-degenerate. Hence it is impossible to change the lengths of interior edges of $\bar{T}$ without changing the curvatures in the first order. Thus P is infinitesimally rigid. $\square$

*Remark* 4.5. In the smooth case, the space of all infinitesimal deformations of a Riemannian metric can be decomposed as a direct sum of conformal, trivial, and anti-conformal deformations. The restriction of the second variation D^2S to the space of conformal deformations is positive definite; trivial deformations don't change the value of S; and on the space of the anti-conformal deformations D^2S is negative definite, provided that the spectrum of the curvature operator satisfies certain assumptions, [4, Chapters 4G, 12H].

In the discrete case, trivial deformations arise from arbitrary displacements of the interior vertices and from displacements of vertices inside the faces orthogonally to those faces. This space has dimension $3i + b$. Conformal deformations should correspond in "blowing up" at each vertex independently, thus their space has dimension i. Thus the signature of D^2S as stated in Theorem 4.3 fits very well with what is known in the smooth case.

Among other works dealing with the signature of the second variation of the discrete Hilbert-Einstein functional let us mention [9, 11].

5. DIRECTIONS FOR FUTURE RESEARCH

Let M be a closed hyperbolic 3-manifold with a geodesic triangulation T. Then the infinitesimal rigidity of M (also known as Calabi-Weil rigidity [20, 7]) is equivalent to dim ker $D^2S = 3i$, where

i is the number of vertices of T. It should be possible to determine the rank of $D^2 S$ (or even better, the signature) by a sort of discrete Bochner method. This would yield a new proof of the Calabi-Weil theorem. A similar method should work for cone-manifolds. If M is a hyperbolic cone-manifold, then M is infinitesimally rigid (in the sense that any deformation preserving the cone-angles is trivial) provided that $\omega_e \leq 2\pi$ around all edges e, [17, 21]; without this assumption M may be infinitesimally flexible, [13].

For ideal triangulations of hyperbolic manifolds, the functional is concave. This makes cusped manifolds the first case to try to reprove the hyperbolization theorem.

The functional is concave also for semiideal triangulations, if all finite vertices lie on the boundary. This was used in [10] to prove the existence of a hyperbolic cusp with a given cone-metric on the boundary. A generalization of this would be realizability of an arbitrary metric with curvature bounded from below by -1 on the boundary of some hyperbolic cusp. On one hand, this should follow from the polyhedral case by an approximation argument; on the other hand, it would be interesting to find a variational proof that uses an extension of the Hilbert-Einstein functional to more general metric spaces. In particular, this is related to the problem at the end of Remark 1.3.

References

[1] Alexander D. Alexandrov. Existence of a convex polyhedron and of a convex surface with a given metric. *Mat. Sbornik, N. Ser.*, 11(53):15–65, 1942. (Russian. English summary).

[2] Michael T. Anderson. Scalar curvature and the existence of geometric structures on 3-manifolds. I. *J. Reine Angew. Math.*, 553:125–182, 2002.

[3] Michael T. Anderson. Scalar curvature and the existence of geometric structures on 3-manifolds. II. *J. Reine Angew. Math.*, 563:115–195, 2003.

[4] Arthur L. Besse. *Einstein manifolds*, volume 10 of *Ergebnisse der Mathematik und ihrer Grenzgebiete (3)*. Springer-Verlag, Berlin, 1987.

[5] Wilhelm Blaschke and Gustav Herglotz. Über die Verwirklichung einer geschlossenen Fläche mit vorgeschriebenem Bogenelement im Euklidischen Raum. *Sitzungsber. Bayer. Akad. Wiss., Math.-Naturwiss. Abt.*, No.2:229–230, 1937.

[6] Alexander I. Bobenko and Ivan Izmestiev. Alexandrov's theorem, weighted Delaunay triangulations, and mixed volumes. *Ann. Inst. Fourier (Grenoble)*, 58(2):447–505, 2008.

[7] Eugenio Calabi. On compact, Riemannian manifolds with constant curvature. I. In *Proc. Sympos. Pure Math., Vol. III*, pages 155–180. American Mathematical Society, Providence, R.I., 1961.

[8] Jeff Cheeger, Werner Müller, and Robert Schrader. On the curvature of piecewise flat spaces. *Comm. Math. Phys.*, 92(3):405–454, 1984.

[9] Daryl Cooper and Igor Rivin. Combinatorial scalar curvature and rigidity of ball packings. *Math. Res. Lett.*, 3(1):51–60, 1996.

[10] François Fillastre and Ivan Izmestiev. Hyperbolic cusps with convex polyhedral boundary. *Geom. Topol.*, 13(1):457–492, 2009.

[11] David Glickenstein. Discrete conformal variations and scalar curvature on piecewise flat two- and three-dimensional manifolds. *J. Differential Geom.*, 87(2):201–237, 2011.

[12] Ivan Izmestiev. The Colin de Verdière number and graphs of polytopes. *Israel J. Math.*, 178:427–444, 2010.

[13] Ivan Izmestiev. Examples of infinitesimally flexible 3–dimensional hyperbolic cone-manifolds. *J. Math. Soc. Japan*, 63(2):581–598, 2011.

[14] Ivan Izmestiev. Infinitesimal rigidity of convex polyhedra through the second derivative of the Hilbert-Einstein functional. *Canad. J. Math.*, 66(4):783–825, 2014.

[15] Ivan Izmestiev and Jean-Marc Schlenker. Infinitesimal rigidity of polyhedra with vertices in convex position. *Pacific J. Math.*, 248(1):171–190, 2010.

[16] Norihito Koiso. Nondeformability of Einstein metrics. *Osaka J. Math.*, 15(2):419–433, 1978.

[17] Rafe Mazzeo and Grégoire Montcouquiol. Infinitesimal rigidity of cone-manifolds and the Stoker problem for hyperbolic and Euclidean polyhedra. *J. Differential Geom.*, 87(3):525–576, 2011.

[18] Tullio Regge. General relativity without coordinates. *Nuovo Cimento*, 19:558–571, 1961.

[19] Stefan Sechelmann. Alexandrov's polyhedron editor, Java Web Start application. http://page.math.tu-berlin.de/ sechel/webstart/AlexandrovPolyhedron.jnlp.

[20] André Weil. On discrete subgroups of Lie groups. *Ann. of Math. (2)*, 72:369–384, 1960.

[21] Hartmut Weiss. The deformation theory of hyperbolic cone-3-manifolds with cone-angles less than 2π. *Geom. Topol.*, 17:329–367, 2013.

[22] Hidehiko Yamabe. On a deformation of Riemannian structures on compact manifolds. *Osaka Math. J.*, 12:21–37, 1960.

Institut für Mathematik, Freie Universität Berlin, Arnimallee 2, 14195 Berlin

Discrete complex analysis – the medial graph approach

Alexander I. Bobenko and Felix Günther

Abstract

We discuss a new formulation of the linear theory of discrete complex analysis on planar quad-graphs based on their medial graphs. It generalizes the theory on rhombic quad-graphs developed by Duffin, Mercat, Kenyon, Chelkak and Smirnov and follows the approach on general quad-graphs proposed by Mercat. We provide discrete counterparts of the most fundamental objects in complex analysis such as holomorphic functions, differential forms, derivatives, and the Laplacian. Also, we discuss discrete versions of important fundamental theorems such as Green's identities and Cauchy's integral formulae. For the first time, Green's first identity and Cauchy's integral formula for the derivative of a holomorphic function are discretized.

1. History

Discrete harmonic functions on the square lattice were studied by a number of authors in the 1920s, including Courant, Friedrichs, and Lewy [5]. Discrete holomorphic functions on the square lattice were studied by Isaacs [10]. He proposed two different definitions for holomorphicity. One of them was reintroduced and further investigated by Lelong-Ferrand [8]. She developed the theory to a level that allowed her to prove the Riemann mapping theorem using discrete methods [13]. Duffin also studied discrete complex analysis on the square grid [6], and he was the first who extended the theory to rhombic lattices [7]. Kenyon [12], and Chelkak and Smirnov [3] resumed the investigation of discrete complex analysis on rhombic lattices, or, equivalently, isoradial graphs.

Mercat extended the theory from domains in the complex plane to discrete Riemann surfaces, first considering cellular decompostions into rhombi [14] and later generalizing the notions to general quadrilaterals [16]. The motivation for this theory of discrete Riemann surfaces is derived from statistical physics, in particular, the Ising model. Mercat defined a discrete Dirac operator and discrete spin structures, and he identifies criticality in the Ising model with rhombic quad-graphs.

Some two-dimensional discrete models in statistical physics exhibit conformally invariant properties in the thermodynamical limit. Such conformally invariant properties were established by Chelkak and Smirnov for the Ising model [4], and by Kenyon for the dimer model on a square grid [11]. In both cases, linear theories of discrete analytic functions on regular grids were highly important. Kenyon, Chelkak and Smirnov obtained important analytic results [12, 3], which were instrumental in the proof that the critical Ising model is universal [4].

Important non-linear discrete theories of complex analysis involve circle packings, or, more generally, circle patterns. Rodin and Sullivan first proved that the Riemann mapping of a complex domain to the unit disk can be approximated by circle packings [17]. A similar result for isoradial

circle patterns, even with irregular combinatorics, is due to Bücking [2]. The first author, Mercat and Suris showed how the linear theory of discrete holomorphic functions on quad-graphs can be obtained by linearizing the theory on circle patterns: Discrete holomorphic functions describe infinitesimal deformations of circle patterns [1].

2. Organization of the paper

Our setup is a strongly regular cellular decompostion of the complex plane into quadrilaterals, called quad-graph, which we assume to be bipartite. Basic notations for quad-graphs used in this paper are introduced in Section 3. Of crucial importance for our work is the medial graph of a quad-graph in Section 4. It provides the connection between the notions of discrete derivatives of Kenyon [12], Mercat [15], and Chelkak and Smirnov [3], extended from rhombic to general quad-graphs, and discrete differential forms and discrete exterior calculus. We discuss the discrete derivatives in Sections 5 and 7. Concerning discrete differential forms in Section 6, we get essentially the same definitions as Mercat proposed in [16]. However, our notation of discrete exterior calculus in Sections 8, 9, and 10 is slightly more general and shows its power when considering integral formulae. In Section 11, we discuss the discrete Laplacian introduced by Mercat [16]. In particular, we prove discrete Green's identities and recover the factorization of the discrete Laplacian known from the rhombic case [12, 15]. We formulate discrete Cauchy's integral formulae for discrete holomorphic functions and their discrete derivatives in Section 12.

To keep the paper short, we highlight just the most instructive proofs, and omit the others. However, the skipped proofs are usually elementary calculations or immediate consequences of previous statements. The proofs and a more detailed discussion of discrete complex analysis on planar quad-graphs can be found in the dissertation of the second author [9]. There, we also investigate discrete Green's functions, prove their existence and the existence of discrete Cauchy's kernels, and provide several results concerning the asymptotics of these functions in the case of certain parallelogram-graphs.

3. Bipartite quad-graphs

We consider a strongly regular and locally finite cellular decompostion of the complex plane $\mathbb{C}$ into quadrilaterals, described by a bipartite *quad-graph* Λ. The sets of vertices, edges, and faces, are denoted by $V(\Lambda)$, $E(\Lambda)$, and $F(\Lambda)$, respectively. We refer to the maximal independent sets of vertices of Λ as *black* and *white* vertices. Let Γ and Γ^* be the graphs defined on the black and white vertices where the edges are exactly the diagonals of faces of Λ. It is easy to see that Γ and Γ^* are dual to each other. For the ease of notation, we identify the vertices of Λ with their corresponding complex values, and to oriented edges of $\Lambda, \Gamma, \Gamma^*$ we assign the complex numbers determined by the difference of their two endpoints.

To Λ we associate its *dual* $\Diamond = \Lambda^*$. In this paper, we look at $\Diamond$ in an abstract way, identifying vertices or faces of $\Diamond$ with corresponding faces or vertices of Λ, respectively. However, in the particular case that all quadrilaterals are parallelograms, it makes sense to place the vertices of $\Diamond$ at the centers of the parallelograms [9]. If a vertex $v \in \Lambda$ is a vertex of a quadrilateral $Q \in \Diamond$, we write $Q \sim v$ or $v \sim Q$ and call v and Q *incident* to each other. The vertices of Q are denoted by b_-, w_-, b_+, w_+ in counterclockwise order, where $b_\pm \in \Gamma$ and $w_\pm \in \Gamma^*$.

Definition 1. For a quadrilateral $Q \in V(\Diamond) \cong F(\Lambda)$ we define

$$\rho(b_-, b_+) = \rho(b_+, b_-) := -i\frac{w_+ - w_-}{b_+ - b_-} =: \frac{1}{\rho(w_-, w_+)} = \frac{1}{\rho(w_+, w_-)}.$$

Let $\varphi_Q := \arccos\left(\mathrm{Re}\left(i\rho(b_-, b_+)\right)\right)$ be the angle under which the diagonal lines of Q intersect.

Figure 3.1 shows a finite bipartite quad-graph together with the notation for a single quadrilateral Q and the *star of a vertex* v, i.e., the set of all faces incident to v.

In addition, we denote by $\Diamond_0$ a connected subset of $\Diamond$. It is called *simply-connected* if the corresponding set of cells in $\mathbb{C}$ is simply-connected. Its vertices induce subgraphs Λ_0 of Λ, Γ_0 of Γ, and Γ_0^* of Γ^*. For simplicity, we always assume that the induced subgraphs are connected as well.
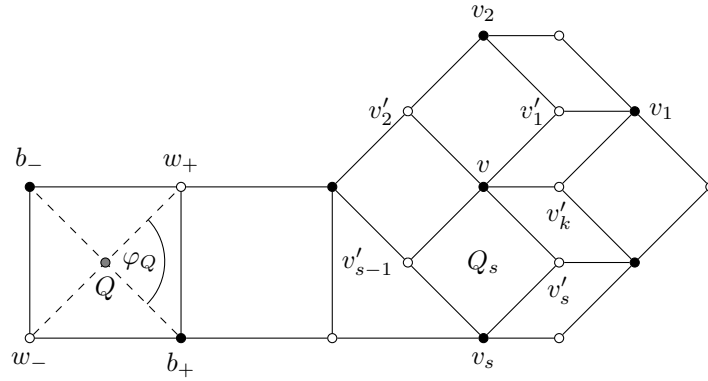
Figure 3.1: Bipartite quad-graph with notations

4. Medial graph

Definition 2. The *medial graph* X of Λ is defined as follows. Its vertex set is given by all the midpoints of edges of Λ, and two vertices are adjacent iff the corresponding edges belong to the same face and have a vertex in common. The set of faces of X is in bijective correspondence with $V(\Lambda) \cup V(\Diamond)$: The vertices of a face F_v corresponding to $v \in V(\Lambda)$ are the midpoints of edges of Λ incident to v, and the vertices of a quadrilateral face F_Q corresponding to $Q \in V(\Diamond)$ are the midpoints of the four edges of Λ belonging to Q.

Any edge e of X is the common edge of two faces F_Q and F_v for $Q \sim v$, denoted by $[Q, v]$.

Let $Q \in V(\Diamond)$ and $v_0 \sim Q$. Due to Varignon's theorem, F_Q is a parallelogram, and the complex number assigned to the edge $e = [Q, v_0]$ connecting the midpoints of edges $v_0 v'_-$ and $v_0 v'_+$ of Λ is just half of $e = v'_+ - v'_-$. In Figure 4.1, showing Λ with its medial graph, the vertices of F_Q and F_v, $v \in V(\Lambda)$, are colored gray.
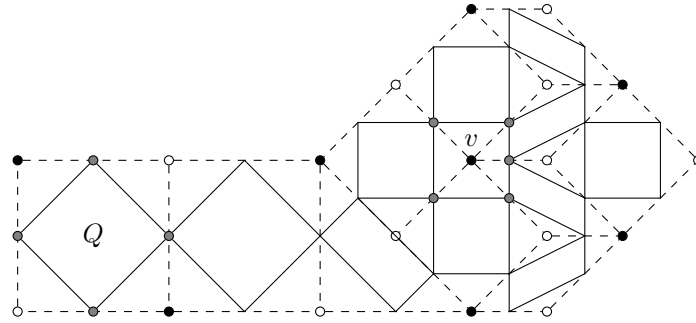
Figure 4.1: Bipartite quad-graph (dashed) with medial graph (solid)

For a subgraph $\Diamond_0 \subseteq \Diamond$, we denote by $X_0 \subseteq X$ the subgraph of X whose edges are contained in faces of $\Diamond_0$. Note that the medial graph X corresponds to a (strongly regular and locally finite) cellular decompostion of $\mathbb{C}$ in a canonical way. In particular, we can talk about a topological disk in $F(X_0)$ and about a (counterclockwise oriented) boundary ∂X_0.

Definition 3. For $v \in V(\Lambda)$ and $Q \in V(\Diamond)$, let P_v and P_Q be the closed paths on X connecting the midpoints of edges of Λ incident to v and Q, respectively, in counterclockwise direction. In Figure 4.1, their vertices are colored gray. We call P_v and P_Q *discrete elementary cycles*.

5. Discrete derivatives of functions on the vertices of the quad-graph

In the classical theory, holomorphic functions (with nowhere-vanishing derivative) preserve angles, and at a single point, lengths are uniformly scaled. This motivates the following definition of discrete holomorphicity [16] that was also used previously in the rhombic setting.

Definition 4. Let $Q \in V(\Diamond)$ and f a complex function on b_-, w_-, b_+, w_+. f is called *discrete holomorphic* at Q if it satisfies the *discrete Cauchy-Riemann equation*

$$\frac{f(b_+) - f(b_-)}{b_+ - b_-} = \frac{f(w_+) - f(w_-)}{w_+ - w_-}.$$

For discrete holomorphicity, only the differences on Γ and Γ^* matter. Hence, we should not consider constants on $V(\Lambda)$, but biconstants [15] determined by each a value on $V(\Gamma)$ and $V(\Gamma^*)$. We call functions that are constant on $V(\Gamma)$ and constant on $V(\Gamma^*)$ *biconstant*.

Definition 5. Let $Q \in V(\Diamond)$, and let f be a complex function on b_-, w_-, b_+, w_+. The *discrete derivatives* $\partial_\Lambda f$, $\bar{\partial}_\Lambda f$ are defined by

$$\partial_\Lambda f(Q) := \frac{\exp\left(-i\left(\varphi_Q - \frac{\pi}{2}\right)\right)}{2\sin(\varphi_Q)} \cdot \frac{f(b_+) - f(b_-)}{b_+ - b_-} + \frac{\exp\left(i\left(\varphi_Q - \frac{\pi}{2}\right)\right)}{2\sin(\varphi_Q)} \cdot \frac{f(w_+) - f(w_-)}{w_+ - w_-},$$

$$\bar{\partial}_\Lambda f(Q) := \frac{\exp\left(i\left(\varphi_Q - \frac{\pi}{2}\right)\right)}{2\sin(\varphi_Q)} \cdot \frac{f(b_+) - f(b_-)}{\overline{b_+ - b_-}} + \frac{\exp\left(-i\left(\varphi_Q - \frac{\pi}{2}\right)\right)}{2\sin(\varphi_Q)} \cdot \frac{f(w_+) - f(w_-)}{\overline{w_+ - w_-}}.$$

In the case of quadrilaterals whose diagonals intersect orthogonally, $\varphi_Q = \pi/2$, and $\partial_\Lambda f, \bar{\partial}_\Lambda f$ are exactly defined as in [3]. They naturally discretize their smooth counterparts $(\partial_x - i\partial_y)/2$ and $(\partial_x + i\partial_y)/2$. In a general quadrilateral Q, we have to take the deviation $(\varphi_z - \pi/2)$ from orthogonality into account, and change the factors appropriately.

Proposition 6. *Let $\Diamond_0 \subseteq \Diamond$ and f be a discrete holomorphic function on $V(\Lambda_0)$.*

(1) *If f is purely imaginary or purely real, f is biconstant.*
(2) *If $\bar{\partial}_\Lambda f \equiv 0 \equiv \partial_\Lambda f$, f is biconstant.*

6. DISCRETE DIFFERENTIAL FORMS

We mainly consider two type of functions, functions $f : V(\Lambda) \to \mathbb{C}$ and functions $h : V(\Diamond) \to \mathbb{C}$. An example for a relevant function on the quadrilateral faces is $\partial_\Lambda f$.

A *discrete one-form* ω is a complex function on the oriented edges of the medial graph X, and a *discrete two-form* Ω is a complex function on the faces of X. The evaluations of ω at an oriented edge e of X and of Ω at a face F of X are denoted by $\int_e \omega$ and $\iint_F \Omega$, respectively.

If P is a directed path of edges $e_1, e_2, \ldots, e_n$ of X, the *discrete integral* along P is defined as $\int_P \omega = \sum_{k=1}^n \int_{e_k} \omega$. For closed paths P, we write $\oint_P \omega$ instead. In the case that P is the boundary of an oriented disk in X, we call it a *discrete contour*. The *discrete integral* of Ω over several faces of X is defined similarly.

Definition 7. The discrete one-forms dz and $d\bar{z}$ are given by $\int_e dz = e$ and $\int_e d\bar{z} = \bar{e}$ for any oriented edge e of X. The discrete two-form Ω_0 is defined by

$$\iint_F \Omega_0 = -4i\,\text{area}(F).$$

Remark 8. Ω_0 is the straightforward discretization of $2dz \wedge d\bar{z}$. It turns out later that several discrete two-forms we are interested in are just defined on half of the faces of X and zero on the other elements of $F(X)$. In order to get results comparable to the classical theory after integration, a factor of two enters in the definitions of Sections 8 and 9. Introducing Ω_0 is a technical trick that allows us to implement this factor of two just in Ω_0. In local coordinates, we can perform our calculations with Ω_0 in the discrete setting exactly as we do with $dz \wedge d\bar{z}$ in the smooth theory, but integration of Ω_0 gives twice the value $dz \wedge d\bar{z}$ yields.

A discrete one-form ω is said to be of *type* $\Diamond$, if for any $Q \in V(\Diamond)$ there exist complex numbers p, q, such that $\omega = pdz + qd\bar{z}$ on all edges $e = [Q, v_s]$, $v_s \sim Q$.

Definition 9. Let $f : V(\Lambda) \to \mathbb{C}$, $h : V(\Diamond) \to \mathbb{C}$, ω a discrete one-form, and Ω a discrete two-form. For any edge $e = [Q, v]$ and any faces F_v, F_Q of X corresponding to the vertex star of $v \in V(\Lambda)$ or

the Varignon parallelogram inside $Q \in V(\Diamond)$, we define the products $f\omega$, $h\omega$, $f\Omega$, and $h\Omega$ by

$$\int_e f\omega := f(v) \int_e \omega \quad \text{and} \quad \iint_{F_v} f\Omega := f(v) \iint_{F_v} \Omega, \quad \iint_{F_Q} f\Omega := 0;$$

$$\int_e h\omega := h(Q) \int_e \omega \quad \text{and} \quad \iint_{F_v} h\Omega := 0, \quad \iint_{F_Q} h\Omega := h(Q) \iint_{F_Q} \Omega.$$

Lemma 10. *Let $Q \in V(\Diamond)$ and f be a complex function on the vertices of Q. Then,*

$$\partial_\Lambda f(Q) = \frac{-1}{4i\,\mathrm{area}(F_Q)} \oint_{P_Q} f d\bar{z} \text{ and } \bar{\partial}_\Lambda f(Q) = \frac{1}{4i\,\mathrm{area}(F_Q)} \oint_{P_Q} f dz.$$

Remark 11. The additional factor of $1/2$ is due to the fact that in analogy to the smooth setup, we should not multiply $f(v)$ with dz (or $d\bar{z}$), but by the arithmetic mean of $f(v)$ and some intermediate value $f(Q)$ instead. Integrating $f dz$ would then eliminate $f(Q)$, so the choice of the intermediate value does not matter.

7. Discrete derivatives of functions on the faces of the quad-graph

Inspired by Lemma 10, we can now define the discrete derivatives for complex functions on $V(\Diamond)$. The reason for the additional factor of $1/2$ remains the same.

Definition 12. Let $v \in V(\Lambda)$ and h be a complex function defined on all quadrilaterals $Q_s \sim v$. Then, the *discrete derivatives* $\partial_\Diamond h$, $\bar{\partial}_\Diamond h$ at v are defined by

$$\partial_\Diamond h(v) := \frac{-1}{4i\,\mathrm{area}(F_v)} \oint_{P_v} h d\bar{z} \text{ and } \bar{\partial}_\Diamond h(v) := \frac{1}{4i\,\mathrm{area}(F_v)} \oint_{P_v} h dz.$$

h is called *discrete holomorphic* at v if $\bar{\partial}_\Diamond h(v) = 0$.

Note that in the rhombic case, our definition coincides with the one in [3]. As an immediate consequence of the definition, we obtain a *discrete Morera's theorem.*

Proposition 13. $f : V(\Lambda) \to \mathbb{C}$ *or* $h : V(\Diamond) \to \mathbb{C}$ *is discrete holomorphic if and only if* $\oint_P f dz = 0$ *or* $\oint_P h dz = 0$, *respectively, for all discrete contours P.*

Definition 14. Let $f_1, f_2 : V(\Lambda) \to \mathbb{C}$ and $h_1, h_2 : V(\Diamond) \to \mathbb{C}$. Their *discrete scalar products* are defined as

$$\langle f_1, f_2 \rangle := -\frac{1}{2i} \iint_{F(X)} f_1 \bar{f}_2 \Omega_0 \quad \text{and} \quad \langle h_1, h_2 \rangle := -\frac{1}{2i} \iint_{F(X)} h_1 \bar{h}_2 \Omega_0,$$

whenever the right hand side converges absolutely.

Note that both discrete two-forms $f_1 \bar{f}_2 \Omega_0$ and $h_1 \bar{h}_2 \Omega_0$ are zero on half of the faces of X, making the factor of two incorporated in Ω_0 necessary.

Proposition 15. $-\partial_\Diamond$ *and* $-\bar{\partial}_\Diamond$ *are the formal adjoints of* $\bar{\partial}_\Lambda$ *and* ∂_Λ, *respectively. That is, if* $f : V(\Lambda) \to \mathbb{C}$ *or* $h : V(\Diamond) \to \mathbb{C}$ *is compactly supported,*

$$\langle \partial_\Lambda f, h \rangle + \langle f, \bar{\partial}_\Diamond h \rangle = 0 = \langle \bar{\partial}_\Lambda f, h \rangle + \langle f, \partial_\Diamond h \rangle.$$

Proof. Using Lemma 10 and $\partial_\Diamond \bar{h} = \overline{\bar{\partial}_\Diamond h}$, we get

$$-2i\langle \partial_\Lambda f, h \rangle - 2i\langle f, \bar{\partial}_\Diamond h \rangle = \sum_{Q \in V(\Diamond)} \overline{h(Q)} \oint_{P_Q} f d\bar{z} + \sum_{v \in V(\Lambda)} f(v) \oint_{P_v} \bar{h} d\bar{z} = \oint_P f \bar{h} d\bar{z} = 0,$$

where P is a large contour such that $f\bar{h}$ vanishes in a neighborhood of P. The second equation is shown in the same way. $\qquad \square$

Remark 16. In the work of Kenyon [12] and Mercat [15] on discrete complex analysis on rhombic quad-graphs, the discrete differentials for functions on the vertices and the faces were constructed in such a way that they are formal adjoints to each other.

As in the rhombic setup [3], the discrete differentials commute in the following way:

Proposition 17. *Let $f : V(\Lambda) \to \mathbb{C}$. Then, $\partial_\Diamond \bar{\partial}_\Lambda f \equiv \bar{\partial}_\Diamond \partial_\Lambda f$. In particular, $\partial_\Lambda f$ is discrete holomorphic if f is discrete holomorphic.*

Remark 18. Note that even in the rhombic case, $\partial_\Lambda \bar{\partial}_\Diamond h \neq \bar{\partial}_\Lambda \partial_\Diamond h$ for generic $h : V(\Diamond) \to \mathbb{C}$ [3].

Proposition 19. *Let $\Diamond_0 \subseteq \Diamond$ be simply-connected. Then, for any discrete holomorphic function h on $V(\Diamond_0)$, there is a discrete primitive $f := \int h$ on $V(\Lambda_0)$, i.e., f is discrete holomorphic and $\partial_\Lambda f = h$. f is unique up to two additive constants on Γ_0 and Γ_0^*.*

Proof. Since h is discrete holomorphic, $\oint_P h\,dz = 0$ for any discrete contour P. Thus, $h\,dz$ can be integrated to a well-defined function f_X on $V(X)$ that is unique up to an additive constant. Using that $h\,dz$ is a discrete one-form of type $\Diamond$, we can construct a function f on $V(\Lambda)$ such that $f_X\left((v+w)/2\right) = (f(v) + f(w))/2$ for any edge (v, w) of Λ. Given f_X, f is unique up to an additive constant.

In summary, f is unique up to two additive constants that can be chosen independently on Γ_0 and Γ_0^*. By construction, f satisfies

$$\frac{f(b_+) - f(b_-)}{b_+ - b_-} = h(Q) = \frac{f(w_+) - f(w_-)}{w_+ - w_-}$$

on any quadrilateral Q. It follows that f is discrete holomorphic and $\partial_\Lambda f = h$. $\qquad \square$

8. Discrete exterior derivative

Our notation of discrete exterior calculus is similar to the approach of Mercat in [14, 15, 16], but differs in some aspects. The main differences are due to our different notation of multiplication of functions with discrete one-forms, which allows us to define a discrete exterior derivative on a larger class of discrete one-forms. It coincides with Mercat's discrete exterior derivative in the case of discrete one-forms of type $\Diamond$. In contrast, our definitions are based on a coordinate representation.

Definition 20. Let $f : V(\Lambda) \to \mathbb{C}$ and $h : V(\Diamond) \to \mathbb{C}$. We define the *discrete exterior derivatives* df and dh as follows:

$$df := \partial_\Lambda f\,dz + \bar{\partial}_\Lambda f\,d\bar{z} \text{ and } dh := \partial_\Diamond h\,dz + \bar{\partial}_\Diamond h\,d\bar{z}.$$

Let ω be a discrete one-form. Around faces F_v and F_Q of X corresponding to vertices $v \in V(\Lambda)$ and $Q \in V(\Diamond)$, respectively, we write $\omega = p\,dz + q\,d\bar{z}$ with functions p, q defined on faces $Q_s \sim v$ or vertices $b_\pm, w_\pm \sim Q$, respectively. The *discrete exterior derivative* $d\omega$ is given by

$$d\omega|_{F_v} := \left(\partial_\Diamond q - \bar{\partial}_\Diamond p\right) \Omega_0 \text{ and } d\omega|_{F_Q} := \left(\partial_\Lambda q - \bar{\partial}_\Lambda p\right) \Omega_0.$$

The reason why we add a factor of two in the definition of $d\omega$ (hidden in Ω_0) is the same as the factor of $1/2$ in the definition of $\partial_\Diamond, \bar{\partial}_\Diamond$: For the definition of $d\omega$, p and q are defined on the vertices of Λ or $\Diamond$, but ω lives halfway between two incident vertices of Λ and $\Diamond$, resulting in the factor of 2.

The representation of ω as $p\,dz + q\,d\bar{z}$ (p, q defined on edges of X) is non-unique, since we represent one complex number as the linear combination of two other complex numbers. However, $d\omega$ is well-defined by *discrete Stokes' theorem*, which also justifies our definition of df and dh.

Lemma 21. *Let $f : V(\Lambda) \to \mathbb{C}$, and let ω be a discrete one-form. Then, for any directed edge e of X starting in the midpoint of the edge vv'_- and ending in the midpoint of the edge vv'_+ of Λ, and for any face F of X with counterclockwise oriented boundary ∂F we have:*

$$\int_e df = \frac{f(v) + f(v'_+)}{2} - \frac{f(v) + f(v'_-)}{2} \text{ and } \iint_F d\omega = \oint_{\partial F} \omega.$$

An easy consequence of the definition of the discrete exterior derivative is that $\iint_F d\omega = 0$ on any face F corresponding to a vertex of Λ, when ω is a discrete one-form of type $\Diamond$. We call a discrete one-form ω *closed*, if $d\omega \equiv 0$. For example, df is closed if f is a complex function on $V(\Lambda)$.

Proposition 22. *Let $f : V(\Lambda) \to \mathbb{C}$. Then, $ddf = 0$.*

Proof. By discrete Stokes' theorem, $ddf = 0$ if $\oint_P df = 0$ for any discrete elementary cycle P. Since df is of type $\Diamond$, the statement is trivially true if $P = P_Q$ for $Q \in V(\Diamond)$. So let $P = P_v$ for $v \in V(\Lambda)$. Using discrete Stokes' theorem again,

$$\oint_{P_v} df = \sum_{Q_s \sim v} \frac{f(v'_s) - f(v'_{s-1})}{2} = 0.$$

$\square$

Remark 23. An analogous statement for functions $h : V(\Diamond) \to \mathbb{C}$ is not true in general, even if h is discrete holomorphic and Λ is a rhombic quad-graph.

Note that Proposition 22 immediately implies Proposition 17 by $ddf = \left(\partial_\Diamond \bar\partial_\Lambda f - \bar\partial_\Diamond \partial_\Lambda f\right) \Omega_0$.

Corollary 24. *Let $f : V(\Lambda) \to \mathbb{C}$. Then, f is discrete holomorphic if and only if $df = pdz$ is closed for some $p : V(\Diamond) \to \mathbb{C}$. In this case, p is discrete holomorphic.*

Corollary 25. *Let $f, g : V(\Lambda) \to \mathbb{C}$ and $h : V(\Diamond) \to \mathbb{C}$.*
   (1) *$fdg + gdf$ is a closed discrete one-form.*
   (2) *If f and h are discrete holomorphic, $fhdz$ is a closed discrete one-form.*

Proof. (1) Let $\omega := fdg + gdf$. By Proposition 22, df and dg are closed. Thus, $\oint_{\partial F} \omega = 0$ for any face F corresponding to $V(\Lambda)$. Using Lemma 10, a direct calculation shows $\oint_{\partial F} \omega = 0$ for any face F corresponding to $V(\Diamond)$. It follows by discrete Stokes' theorem that $d\omega = 0$.
   (2) By discrete Morera's theorem, $\oint_{\partial F} fhdz = 0$ for any face F of X, so $fhdz$ is closed. $\square$

Remark 26. In particular, a product $f \cdot g : V(X) \to \mathbb{C}$ can be defined by integration, and $f \cdot g$ is defined up to an additive constant. Furthermore, $f \cdot h : E(X) \to \mathbb{C}$ can be defined by "pointwise" multiplication. If all these functions are holomorphic, $fdg + gdf = pdz$ is closed ($p : E(X) \to \mathbb{C}$) and so to say a discrete holomorphic one-form, meaning that $f \cdot g$ is discrete holomorphic in this sense. Similarly, $fhdz$ is closed, so $f \cdot h$ is kind of discrete holomorphic by a discrete Morera's theorem. However, $f \cdot g$ and $f \cdot h$ are generally not discrete holomorphic everywhere according to the classical quad-based definition of discrete holomorphicity on the dual of a bipartite quad-graph [9].

9. Discrete wedge product

Following Whitney [19], Mercat defined in [14] a discrete wedge product for discrete one-forms living on the edges of Λ. Then, the discrete exterior derivative defined by a discretization of Stokes' theorem is a derivation for the discrete wedge product. However, a discrete Hodge star cannot be defined on Λ. To circumvent this problem, Mercat used an averaging map to relate discrete one-forms on the edges of Λ with discrete one-forms on the edges of Γ and Γ^*, i.e., discrete one-forms of type $\Diamond$. Then, he could define a discrete Hodge star; however, the discrete exterior derivative was not a derivation for the now heteregoneous discrete wedge product anymore.

We propose a different interpretation of the discrete wedge product. It the end, we somehow recover the definitions Mercat proposed in [14, 15, 16], but our derivation is different. Starting with discrete one-forms of type $\Diamond$ that are defined on the edges of X, we obtain a discrete wedge product on the faces of X that vanishes on half of the faces. This definition is different from Whitney's [19] and has the advantage that both a discrete wedge product and a discrete Hodge star can be defined on the same structure. In contrast to Mercat's work, we now can make sense out of the statement that the discrete exterior derivative is a derivation for the discrete wedge product, see Proposition 29. This proposition is of crucial importance to deduce discrete integral formulae such as discrete Green's identities.

Lemma 27. *Let ω be a discrete one-form of type $\Diamond$. Then, there is a unique representation $\omega = pdz + qd\bar{z}$ with functions $p, q : V(\Diamond) \to \mathbb{C}$. On a quadrilateral $Q \in V(\Diamond)$, p and q are given by*

$$p(Q) = \frac{1}{2\sin(\varphi_Q)} \left(\exp\left(-i\left(\varphi_Q - \frac{\pi}{2}\right)\right) \frac{\int_e \omega}{e} + \exp\left(i\left(\varphi_Q - \frac{\pi}{2}\right)\right) \frac{\int_{e^*} \omega}{e^*} \right),$$

$$q(Q) = \frac{1}{2\sin(\varphi_Q)} \left(\exp\left(i\left(\varphi_Q - \frac{\pi}{2}\right)\right) \frac{\int_e \omega}{\bar{e}} + \exp\left(-i\left(\varphi_Q - \frac{\pi}{2}\right)\right) \frac{\int_{e^*} \omega}{\bar{e}^*} \right).$$

Here, e is an edge of X parallel to a (black) edge of Γ, and e^* corresponds to an (white) edge of Γ^*.

Definition 28. Let $\omega = p\,dz + q\,d\bar{z}$ and $\omega' = p'\,dz + q'\,d\bar{z}$ be two discrete one-forms of type $\lozenge$, $p, p', q, q' : V(\lozenge) \to \mathbb{C}$ given by Lemma 27. Then, the *discrete wedge product* $\omega \wedge \omega'$ is defined as the discrete two-form being 0 on faces of X corresponding to vertices of Λ that equals

$$(pq' - qp')\,\Omega_0$$

on faces corresponding to $V(\lozenge)$.

By definition, the discrete wedge product vanishes on faces of X corresponding to $V(\Lambda)$. Since the faces of X corresponding to $V(\lozenge)$ cover exactly half of the area of the quadrilaterals, the factor of two in the definition of Ω_0 compared to $dz \wedge d\bar{z}$ incorporates the vanishing regions of the discrete wedge product.

Proposition 29. *If $f : V(\Lambda) \to \mathbb{C}$ and ω is a discrete one-form of type $\lozenge$, $d(f\omega) = df \wedge \omega + f\,d\omega$.*

Proof. Let $\omega = p\,dz + q\,d\bar{z}$ with $p, q : V(\lozenge) \to \mathbb{C}$ given by Lemma 27. For $v \in V(\Lambda)$ and $Q \in V(\lozenge)$,

$$d(f\omega)|_{F_v} = \big(f(v)\,(\partial_\lozenge q)\,(v) - f(v)\,(\bar{\partial}_\lozenge p)\,(v)\big)\,\Omega_0 = f\,d\omega|_{F_v},$$
$$d(f\omega)|_{F_Q} = \big(q(Q)\,(\partial_\Lambda f)\,(Q) - p(Q)\,(\bar{\partial}_\Lambda f)\,(Q)\big)\,\Omega_0 = (df \wedge \omega))|_{F_Q}.$$

But $(df \wedge \omega)|_{F_v} = 0$ and $f\,d\omega|_{F_Q} = 0$, so $d(f\omega) = df \wedge \omega + f\,d\omega$. $\qquad\square$

10. Discrete Hodge star

Definition 30. Let $f : F(\Lambda) \to \mathbb{C}$, $h : V(\lozenge) \to \mathbb{C}$, $\omega = p\,dz + q\,d\bar{z}$ a discrete one-form of type $\lozenge$ with complex functions $p, q : V(\lozenge) \to \mathbb{C}$ given by Lemma 27, and Ω a discrete two-form. The *discrete Hodge star* is given by

$$\star f := -\frac{1}{2i} f\Omega_0; \quad \star h := -\frac{1}{2i} h\Omega_0; \quad \star\omega := -ip\,dz + iq\,d\bar{z}; \quad \star\Omega := -2i\frac{\Omega}{\Omega_0}.$$

If ω and ω' are both discrete one-forms of type $\lozenge$, we define their *discrete scalar product*

$$\langle \omega, \omega' \rangle := \iint_{F(X)} \omega \wedge \star\bar{\omega}',$$

whenever the right hand side converges absolutely. Similarly, a discrete scalar product for discrete two-forms is defined.

Note that $\star\Omega$ is a priori a function on $F(X)$. However, the discrete two-forms to that we will apply the discrete Hodge star vanish on all faces of X corresponding to faces of Λ or on all faces corresponding to vertices of Λ. In these cases, $\star\Omega$ is a function on $V(\Lambda)$ or on $V(\lozenge)$, respectively.

Corollary 31.   (1) $\star^2 = \mathrm{Id}$ *on complex functions on $V(\Lambda)$ or $V(\lozenge)$ and discrete two-forms.*
  (2) $\star^2 = -\mathrm{Id}$ *on discrete one-forms of type $\lozenge$.*
  (3) $\langle f_1, f_2 \rangle = \iint_{F(X)} f_1 \overline{\star f_2}$ *and* $\langle h_1, h_2 \rangle = \iint_{F(X)} h_1 \overline{\star h_2}$ *for functions $f_1, f_2 : V(\Lambda) \to \mathbb{C}$ and $h_1, h_2 : V(\lozenge) \to \mathbb{C}$.*
  (4) $f : V(\Lambda) \to \mathbb{C}$ *is discrete holomorphic if and only if $\star df = -idf$.*

Remark 32. It can be easily checked that our definition of a discrete Hodge star on discrete one-forms coincides with Mercat's definition given in [16]. But on discrete two-forms and complex functions, our definition of the discrete Hodge star includes an additional factor of the area of the corresponding face of X. As before, the additional factor of two encoded in Ω_0 reflects the fact that the corresponding two-forms vanish on half of the faces of X.

Proposition 33. $\delta := -\star d\star$ *is the formal adjoint of the discrete exterior derivative d: Let $f : V(\Lambda) \to \mathbb{C}$, ω a discrete one-form of type $\lozenge$, and Ω a discrete two-form being 0 on all faces corresponding to vertices of $\lozenge$. Assume that all of them are compactly supported. Then,*

$$\langle df, \omega \rangle = \langle f, \delta\omega \rangle \text{ and } \langle d\omega, \Omega \rangle = \langle \omega, \delta\Omega \rangle.$$

Proof. Using discrete Stokes' theorem, Proposition 29, and Corollary 31 (i), we obtain

$$0 = \iint\limits_{F(X)} d(f \star \bar{\omega}) = \iint\limits_{F(X)} df \wedge \star \bar{\omega} + \iint\limits_{F(X)} f d \star \bar{\omega} = \langle df, \omega \rangle + \langle f, \star d \star \omega \rangle.$$

The second equation is shown in the same manner. □

11. Discrete Laplacian

The discrete Laplacian and the discrete Dirichlet energy on general quad-graphs were first introduced by Mercat in [16]. Later, Skopenkov reintroduced these definitions in [18], taking the same definition in a different notation.

Definition 34. The *discrete Laplacian* on discrete differential forms is defined as the operator

$$\triangle := -\delta d - d\delta = \star d \star d + d \star d \star.$$

A function $f : V(\Lambda) \to \mathbb{C}$ is called *discrete harmonic* at $v \in V(\Lambda)$ if $\triangle f(v) = 0$.

The following factorization of the discrete Laplacian in terms of discrete derivatives generalizes the corresponding results given in [3] to general quad-graphs. The local representation of $\triangle f$ at $v \in V(\Lambda)$ is, up to a factor involving the area of the face F_v, the same as in [16].

Corollary 35. *Let $f : V(\Lambda) \to \mathbb{C}$. Then, $\triangle f = 4\partial_\Diamond \bar{\partial}_\Lambda f = 4\bar{\partial}_\Diamond \partial_\Lambda f$. At a vertex v of Λ,*

$$\triangle f(v) = \frac{1}{4\text{area}(F_v)} \sum_{Q_s \sim v} \frac{1}{\text{Re}\,(\rho(v, v_s))} \left(|\rho(v, v_s)|^2 \,(f(v_s) - f(v)) + \text{Im}\,(\rho(v, v_s))\,(f(v_s') - f(v_{s-1}')) \right).$$

Remark 36. In the case that the diagonals of the quadrilaterals are orthogonal to each other, ρ is always real. Then, the discrete Laplacian splits into two discrete Laplacians on Γ and Γ^*.

Corollary 37. *Let $f : V(\Lambda) \to \mathbb{C}$.*

(1) *If f is discrete harmonic, $\partial_\Lambda f$ is discrete holomorphic.*
(2) *If f is discrete holomorphic, f, $\text{Re}\,f$, and $\text{Im}\,f$ are discrete harmonic.*

For a finite subset $\Diamond_0 \subset \Diamond$ and two functions $f, g : V(\Lambda_0) \to \mathbb{C}$, we denote by

$$\langle f, g \rangle_{\Diamond_0} := -\frac{1}{2i} \iint\limits_{F(X_0)} f \bar{g} \Omega_0$$

the discrete scalar product of f and g restricted to $\Diamond_0$. Similarly, the restriction of the discrete scalar product of two discrete one-forms is defined.

In the rhombic setup, discrete versions of Green's second identity were already stated by Mercat [14], whose integrals were not well defined separately, and Chelkak and Smirnov [3], whose boundary integral was an explicit sum involving boundary angles. We are able to provide a *discrete Green's first identity* out of which *discrete Green's second identity* immediately follows. The formulation and the proof is a complete analog to the smooth setting.

Theorem 38. *Let $\Diamond_0 \subset \Diamond$ be finite, and let $f, g : V(\Lambda_0) \to \mathbb{C}$.*

(1) $\langle f, \triangle g \rangle_{\Diamond_0} + \langle df, dg \rangle_{\Diamond_0} = \oint\limits_{\partial X_0} f \star d\bar{g}$.
(2) $\langle \triangle f, g \rangle_{\Diamond_0} - \langle f, \triangle g \rangle_{\Diamond_0} = \oint\limits_{\partial X_0} (f \star d\bar{g} - \bar{g} \star df)$.

Proof. By Proposition 29, $d(f \star d\bar{g}) = df \wedge \star d\bar{g} + f \star (\star d \star d\bar{g})$. Now, discrete Stokes' theorem yields the desired result. For the second part, just apply twice discrete Green's first identity. □

12. Discrete Cauchy's integral formulae

Definition 39. Functions $K_{Q_0} : V(\Lambda) \to \mathbb{C}$ and $K_{v_0} : V(\Diamond) \to \mathbb{C}$ are called *discrete Cauchy's kernels* (*with respect to $Q_0 \in V(\Diamond)$ or $v_0 \in V(\Lambda)$,* respectively), if for all $Q \in V(\Diamond), v \in V(\Lambda)$ there holds:

$$\bar{\partial}_\Lambda K_{Q_0}(Q) = \delta_{QQ_0} \frac{\pi}{2\text{area}(F_Q)} \quad \text{and} \quad \bar{\partial}_\Diamond K_{v_0}(v) = \delta_{vv_0} \frac{\pi}{2\text{area}(F_v)}.$$

Remark 40. In the general case, it seems to be practically impossible to speak about any asymptotic behavior of certain functions, as Kenyon did for discrete Green's functions and discrete Cauchy's kernels on rhombic quad-graphs [12]. For this reason, we do not require any asymptotic behavior of discrete Cauchy's kernels. However, we can construct discrete Cauchy's kernels on parallelogram graphs with appropriate asymptotics and can prove at least existence of discrete Cauchy's kernels with respect to $Q_0 \in V(\lozenge)$ or $v_0 \in V(\Lambda)$ in the general case [9].

Theorem 41. *Let f and h be discrete holomorphic functions on $V(\Lambda)$ and $V(\lozenge)$, respectively. Let $v_0 \in V(\Lambda)$ and $Q_0 \in V(\lozenge)$, and let $K_{v_0} : V(\lozenge) \to \mathbb{C}$ and $K_{Q_0} : V(\Lambda) \to \mathbb{C}$ be discrete Cauchy's kernels with respect to v_0 and Q_0, respectively.*

Then, for any discrete contours C_{v_0} and C_{Q_0} on X surrounding v_0 and Q_0 once in counterclockwise order, respectively, discrete Cauchy's integral formulae are true:

$$f(v_0) = \frac{1}{2\pi i} \oint_{C_{v_0}} f K_{v_0} dz \text{ and } h(Q_0) = \frac{1}{2\pi i} \oint_{C_{Q_0}} h K_{Q_0} dz.$$

Remark 42. In the case of rhombic quad-graphs, Mercat formulated a discrete Cauchy's integral formula for the average of a discrete holomorphic function on $V(\Lambda)$ along an edge of Λ. In [3], Chelkak and Smirnov provided a discrete Cauchy's integral formula for discrete holomorphic functions on $V(\lozenge)$ using two integrals along cycles on Γ and Γ^*.

Theorem 43. *Let $f : V(\Lambda) \to \mathbb{C}$ be discrete holomorphic, $Q_0 \in V(\lozenge)$, and let $K_{Q_0} : V(\Lambda) \to \mathbb{C}$ be a discrete Cauchy's kernel with respect to Q_0.*

Then, for any discrete contour C_{Q_0} in X surrounding Q_0 once in counterclockwise order that does not contain any edge inside Q_0, the discrete Cauchy's integral formula is true:

$$\partial_\Lambda f(Q_0) = -\frac{1}{2\pi i} \oint_{C_{Q_0}} f \partial_\Lambda K_{Q_0} dz.$$

Proof. Let D be the discrete domain in X bounded by C_{Q_0}. Since no edge of C_{Q_0} passes through Q_0, the discrete one-form $\bar{\partial} K_{Q_0} d\bar{z}$ vanishes on C_{Q_0}. Therefore,

$$\oint_{C_{Q_0}} f \partial_\Lambda K_{Q_0} dz = \oint_{C_{Q_0}} f dK_{Q_0} = \iint_D d(f dK_{Q_0}) = \iint_D df \wedge dK_{Q_0}$$

due to discrete Stokes' theorem, and Propositions 22 and 29. Now, f is discrete holomorphic, so $df \wedge dK_{Q_0} = \partial_\Lambda f \bar{\partial}_\Lambda K_{Q_0} \Omega_0$. But $\bar{\partial}_\Lambda K_{Q_0}$ vanishes on all vertices of $\lozenge$ but Q_0. Finally,

$$-\frac{1}{2\pi i} \oint_{C_{Q_0}} f \partial_\Lambda K_{Q_0} dz = -\frac{1}{2\pi i} \iint_{F_{Q_0}} \partial_\Lambda f \bar{\partial}_\Lambda K_{Q_0} \Omega_0 = \partial_\Lambda f(Q_0).$$

$\square$

REFERENCES

[1] A.I. Bobenko, C. Mercat, and Yu.B. Suris. Linear and nonlinear theories of discrete analytic functions. Integrable structure and isomonodromic Green's function. *J. Reine Angew. Math.*, 583:117–161, 2005.

[2] U. Bücking. Approximation of conformal mappings by circle patterns. *Geom. Dedicata*, 137:163–197, 2008.

[3] D. Chelkak and S. Smirnov. Discrete complex analysis on isoradial graphs. *Adv. Math.*, 228:1590–1630, 2011.

[4] D. Chelkak and S. Smirnov. Universality in the 2D Ising model and conformal invariance of fermionic observables. *Invent. Math.*, 189(3):515–580, 2012.

[5] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differentialgleichungen der mathematischen Physik. *Math. Ann.*, 100:32–74, 1928.

[6] R.J. Duffin. Basic properties of discrete analytic functions. *Duke Math. J.*, 23(2):335–363, 1956.

[7] R.J. Duffin. Potential theory on a rhombic lattice. *J. Comb. Th.*, 5:258–272, 1968.

[8] J. Ferrand. Fonctions préharmoniques et fonctions préholomorphes. *Bull. Sci. Math. Ser. 2*, 68:152–180, 1944.

[9] F. Günther. *Discrete Riemann surfaces and integrable systems*. PhD thesis, Technische Universität Berlin, September 2014. `http://opus4.kobv.de/opus4-tuberlin/files/5659/guenther_felix.pdf`.

[10] R.Ph. Isaacs. A finite difference function theory. *Univ. Nac. Tucumán. Rev. A*, 2:177–201, 1941.

[11] R. Kenyon. Conformal invariance of domino tiling. *Ann. Probab.*, 28(2):759–795, 2002.

[12] R. Kenyon. The Laplacian and Dirac operators on critical planar graphs. *Invent. math.*, 150:409–439, 2002.

[13] J. Lelong-Ferrand. *Représentation conforme et transformations à intégrale de Dirichlet bornée*. Gauthier-Villars, Paris, 1955.

[14] C. Mercat. Discrete Riemann surfaces and the Ising model. *Commun. Math. Phys.*, 218(1):177–216, 2001.

[15] C. Mercat. Discrete Riemann surfaces. In *Handbook of Teichmüller theory. Vol. I*, volume 11 of *IRMA Lect. Math. Theor. Phys.*, pages 541–575, Zurich, 2007. Eur. Math. Soc.

[16] C. Mercat. Discrete complex structure on surfel surfaces. In *Proceedings of the 14th IAPR international conference on Discrete geometry for computer imagery*, DGCI'08, pages 153–164, Berlin, Heidelberg, 2008. Springer-Verlag.

[17] B. Rodin and D. Sullivan. The convergence of circle packings to the Riemann mapping. *J. Diff. Geom.*, 26(2):349–360, 1987.

[18] M. Skopenkov. The boundary value problem for discrete analytic functions. *Adv. Math.*, 240:61–87, 2013.

[19] H. Whitney. Product on complexes. *Ann. Math.*, 39(2):397–432, 1938.

Institut für Mathematik, MA 8-4, Technische Universität Berlin, Straße des 17. Juni 136, 10623 BERLIN, GERMANY • bobenko@math.tu-berlin.de • fguenth@math.tu-berlin.de

Multigrid-convergence of digital curvature estimators

Jacques-Olivier Lachaud

Abstract

Many methods have been proposed to estimate differential geometric quantities like curvature(s) on discrete data. A common characteristics is that they require (at least) one user-given scale or window parameter, which smoothes data to take care of both the sampling rate and possible perturbations. Digital shapes are specific discrete approximation of Euclidean shapes, which come from their digitization at a given grid step. They are thus subsets of the digital plane $\mathbb{Z}^d$. A digital geometric estimator is called *multigrid convergent* whenever the estimated quantity tends towards the expected geometric quantity as the grid step gets finer and finer. The problem is then: can we define curvature estimators that are multigrid convergent without such user-given parameter ? If so, what speed of convergence can we achieve ? We review here three digital curvature estimators that aim at this objective: a first one based on maximal digital circular arc, a second one using a global optimization procedure, a third one that is a digital counterpart to integral invariants and that works on 2D and 3D shapes. We close the exposition by a discussion about their respective properties and their ability to measure curvatures on gray-level images.

1. Introduction

Context and objectives. In many shape processing applications, the estimation of differential quantities on the shape boundary is usually an important step. Their correct estimation makes easier further processing, like quantitative evaluation, feature detection, shape matching or visualization. A considerable amount of approaches have been proposed to estimate curvature(s) given only discrete data. It is often desirable to have theoretical guarantees on the given estimation. This property is called *stability* in Geometry processing: given a continuous shape and a specific sampling of its boundary, the estimated measure should converge to the Euclidean one when the sampling become denser. Perhaps Amenta *et al.* [2] is one of the first work toward this goal.

When discrete data are **meshes**, most approaches are local and do not provide theoretical guarantees (see [41] and [18] for comprehensive evaluations, and Desbrun *et al.* [13] or Bobenko and Suris [3] for a more general theory). Results on the stability of curvature estimators are scarce. We may quote the result [42] for Gaussian curvature, integral curvature measures [10, 11], and to some extent integral invariants of [35, 34].

When discrete data are limited to **point clouds**, fitting polynomials is probably the most common approach (e.g. the osculating jets of Cazals and Pouget [5] is representative of these approaches), but the stability result is restricted to a perfect sampling. A more appealing family of techniques exploits the Voronoi diagram [1, 30, 31]. Several stability results are achieved even in presence of (Hausdorff noise), but they do not entail the stability of curvature estimations.

Note that all the preceding approaches require some parameter tuning, the most important one determining the window of computation or the *scale* of the estimation in the terminology of scale-spaces.

This paper focuses on estimating the curvature (or curvature tensor) on the boundary of **digital shapes**. Such digital structures are subsets of the d-dimensional digital space $\mathbb{Z}^d$ and come generally from the digitization of some Euclidean shape. Of course, the curvature tensor estimation should be as close as possible to the curvature tensor of the underlying Euclidean shape before digitization. Digital data form a special case of discrete data with specific properties: (1) digital data cannot sample the boundary of the Euclidean shape (i.e. they do not lie on the shape boundary), (2) digital data are distributed around the true sample according to arithmetic noise, which looks rather uniform over a range $[-h, h]$ from a statistical point of view, where h is the digitization grid step. Another (weaker) way of stating these characteristics is to say that the Hausdorff distance between the Euclidean shape and its digitization is some $O(h)$. Of course, the quality of the estimation should be improved as the digitization step gets finer and finer. This property is called the *multigrid convergence* [22, 9]. It is thus similar in spirit with the *stability* property.

For 2D digital objects, a few approaches achieve multigrid convergence with some hypotheses. We quote the ones based either on binomial convolution principles [29, 15]. Algorithms are parametrized by the size of the support of the convolution kernel. Convergence theorem holds when such support size is an increasing function of the grid resolution and some shape characteristics. The polynomial fitting method of [36] is an almost parameter-free method for estimating second derivatives on functional digital data, and could perhaps be adapted to estimate the curvature along 2D contours. For 3D digital objects, several empirical methods exist for estimating curvatures, but none achieves multigrid convergence (e.g. see [27, 17]).

We look for digital curvature estimators with the following properties: (1) provably uniformly multigrid convergent, (2) accurate in practice, (3) computable in an exact manner, (4) efficiently evaluable at one point or everywhere, (5) robust to perturbations (i.e. bad digitization around the boundary, outliers), (6) parameter free. The last point is crucial since it allows the analysis of shapes *without any user supervision*. Note that parameter free convergence results have been obtained for length [40, 23] and normal vector estimation [24, 26, 12].

Paper organization. We review here three different approaches which aim at fulfilling these goals:

> **2D:** Maximal Digital Circular Arcs (Section 3),
> **2D/3D:** Constrained minimization of squared curvature (Section 4),
> **2D/3D:** digital Integral Invariants (Section 5),

and we discuss there respective qualities in the last part. Note that the presentation of the different estimators may slightly differ from the original papers. Indeed, the intent is to homogenize notations and properties.

2. Notations and preliminaries

Shapes, digitization, boundary. In all subsequent sections, the symbol $\mathbb{X}$ denotes a family of compact simply connected subsets of $\mathbb{R}^d$ with continuous curvature fields. The *Gauss digitization* $\mathrm{Dig}_h(X)$ of $X \in \mathbb{X}$ with grid step h is defined as the set of integer points within the dilation of X by a factor $\frac{1}{h}$, i.e. $\mathrm{Dig}_h(X) \overset{def}{=} (\frac{1}{h} \cdot X) \cap \mathbb{Z}^d$. Any finite subset Z of $\mathbb{Z}^d$ is called a *digital shape*. Its *digital boundary* $\Delta(Z)$ is the set of $d-1$-dimensional cubical cells that form the topological border of $\cup_{z \in Z} \mathcal{Q}_z$, where $\mathcal{Q}_z$ is the unit cube centered on z. The *h-boundary* $\partial_h X$ of a shape X is the union of the cells of the digital boundary of $\mathrm{Dig}_h(X)$, rescaled by a factor h, i.e.

$$\partial_h X \overset{def}{=} h \cdot \cup_{c \in \Delta(\mathrm{Dig}_h(X))} c = \partial \cup_{z \in \mathrm{Dig}_h(X)} \mathcal{Q}_z.$$

Some of these notions are illustrated on Fig. 5.1, right.

Digital contour, multigrid convergence. Geometrically, the h-boundary of X is close (in the Hausdorff sense) to the topological boundary of X, but it is combinatorially equivalent to the digital boundary of the digitization of X with step h. In 2D, the digital boundary is often called *digital contour*, since it is easy to organize its 1-cells as one or several sequences of 1-cells (called

linels or *cracks* depending on authors). Of course, a *pixel* is a point of $\mathbb{Z}^2$ and *voxel* is a point of $\mathbb{Z}^3$. The multigrid convergence property for local geometric estimators is formally defined below.

Definition 1. The estimator $\hat{\kappa}$ is *multigrid-convergent* toward the geometric quantity κ for the family $\mathbb{X}$ if and only if, for any $X \in \mathbb{X}$, there exits some positive h_0 such that, for any $0 < h < h_0$,

$$\forall x \in \partial X, \forall y \in \partial_h X \text{ with } \|y - x\|_1 \leq h, |\hat{\kappa}(\text{Dig}_h(X), y, h) - \kappa(X, x)| \leq \tau_x(h),$$

where $\tau_{X,x} : \mathbb{R}^{+*} \to \mathbb{R}^+$ has null limit at 0. This function defines the speed of convergence of $\hat{\kappa}$ toward κ at point x of X. The convergence is *uniform* for X when every $\tau_{X,x}$ is bounded from above by a function τ_X independent of $x \in X$ with null limit at 0.

Medial axis, projection, and reach. For a compact set $X \subset \mathbb{R}^d$, let δ_X be the distance function to ∂X. The *medial axis $MA_{\partial X}$* of ∂X is the subset of $\mathbb{R}^d$ whose points have at least two closest points on ∂X. Any point x of $\mathbb{R}^d \setminus \text{MA}_{\partial X}$ has only one closest point on ∂X which we denote by $\xi_X(x)$. The mapping ξ_X is called *projection* and is defined for almost every point of $\mathbb{R}^d$. The *reach of ∂X* [16] is the infimum of $\{\delta_X(y), y \in \text{MA}_{\partial X}\}$. It is denoted by $\rho_{\partial X}$. Note that any shape of $\mathbb{X}$ has a positive reach, which is related to the inverse of the maximal curvature but also to the gaps between shape parts.

3. Curvature by maximal digital circular arcs

The curvature estimator by maximal digital circular arcs (MDCA) was introduced in [38]. Maximal digital straight segments proved to be an excellent basis for tangent estimation. Hence maximal digital circular arcs is an excellent candidate for curvature estimation.

Let C be some digital contour to a digital shape Z. We look only at connected contours, since each connected component can be treated separately. In this case, the digital contour is a circular sequence of linels. Any proper connected part C' of C is a sequence of linels, whose *discrete length* is its number of linels. Each linel of C' lies between two edge-adjacent pixels, one pixel belonging to Z and called *interior* to C', the other pixel belonging to $\mathbb{Z}^2 \setminus Z$ and called *exterior* to C'.

Any part C' of C is a *digital circular arc* (DCA for short) if and only if the interior and exterior pixels of C' are circularly separable, i.e. there exists a (Euclidean) circle that either encloses the interior points without enclosing any exterior points or that encloses the exterior points without enclosing any interior points. Any map associating to a DCA A the value 0 if the interior and exterior points of A are linearly separable and the curvature of an arbitrary separating circle otherwise is denoted by k.

Any DCA A of C is *maximal* if and only if all the parts C' containing A, i.e. such that $A \subsetneq C' \subset C$, are not a DCA. The set of all maximal DCA (MDCA for short) that lie on a given contour is unique. Two distinct MDCA have two distinct starting linels and two distinct ending linels. The MDCA can be ordered according to the position of their first linel in the contour. Let us then denote by $(A_i)_{i \in \{1,\dots,n\}}$ the sequence of the n MDCA lying on C.

As a result, a contour C can be partitioned without ambiguity into a sequence $(V_i)_{i \in \{1,\dots,n\}}$ such that V_i is the set of linels closer to $m(A_i)$ than to any other linel $m(A_j)$, $j \in \{1, \dots, n\}$ and $j \neq i$ (the first one with respect to the clockwise orientation of the contour is assumed to be closer in case of tie).

Definition 2. Let $Z \subset \mathbb{Z}^2$ be a digital shape of digital contour $C = \Delta(Z)$. Let p be any point of a linel $c \in C$. Then linel c and thus point p belongs to some V_i. The *parameter-free MDCA curvature estimator* $\hat{\kappa}^\star_{\text{MDCA}}$ is defined as

$$(3.1) \qquad \hat{\kappa}^\star_{\text{MDCA}}(Z, p) \stackrel{def}{=} k(A_i).$$

The *rescaled MDCA curvature estimator* $\hat{\kappa}_{\text{MDCA}}$ is naturally defined for some point $\hat{x} \in \partial_h X$ as

$$(3.2) \qquad \hat{\kappa}_{\text{MDCA}}(\text{Dig}_h(X), \hat{x}, h) \stackrel{def}{=} \frac{1}{h}\hat{\kappa}^\star_{\text{MDCA}}(\text{Dig}_h(X), \frac{1}{h}\hat{x}).$$

This estimator approaches the curvature at a pointel as the curvature of the most-centered maximal digital circular arc around it (see Figure 3.1).

We have a limited multigrid convergence result for the MDCA estimator, whose validity depends on the asymptotic length of maximal digital circular arcs.
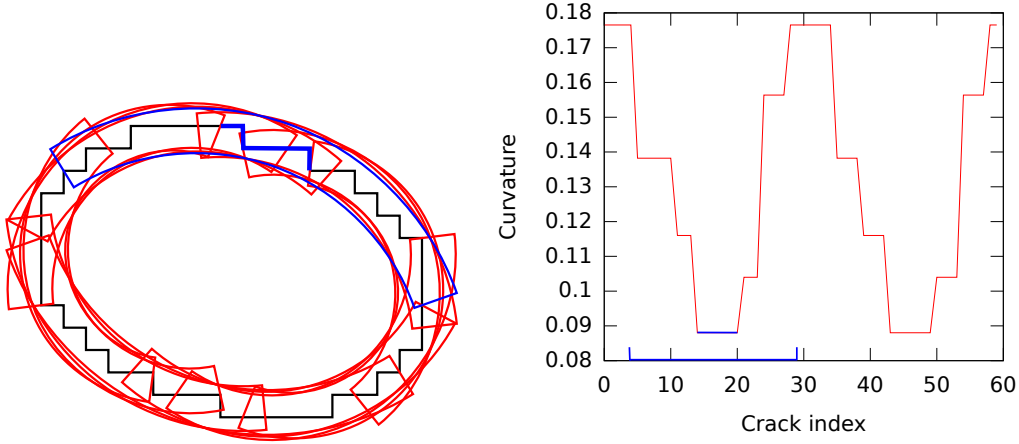
Figure 3.1: The set of MDCAs (12 arcs) is depicted in a) with pieces of rings along the contour of the digitization of an ellipse having a great axis of 9 pixels long and a small axis of 6 pixels long. The angle between the main orientation and the x-axis is equal to 1.9 radians. The curvature plot defined from the set of MDCAs is shown in b). The blue grid edges are those whose curvature depends on the radius of the blue MDCA.

Theorem 3 (Theorem 1, [38]). *Let $X \in \mathbb{X}$. If the Euclidean length of MDCAs along any $\partial_h X$ is lower bounded by $\Omega(h^a)$ and upper bounded by $O(h^b)$, $0 < b \leq a < 1/2$, then the curvature estimator $\hat{\kappa}_{MDCA}$ is uniformly multigrid convergent to the curvature κ, with $\tau = O(h^{\min(1-2a,b)})$.*

Although experiments indicate that the length of MDCA falls into the hypothesis of this theorem, this fact is not proven. However, experiments show that this estimator is very accurate and convergent in practice.

4. CURVATURE BY MINIMIZATION OF SQUARED CURVATURE

A completely different approach to curvature estimation was proposed in [19, 20]. Given a digital shape Z, the idea is to take into account *all* the smooth Euclidean shapes whose digitization is Z. Then, among all these shapes, the most representative shape for curvature estimation is the one that minimizes its total squared curvature. More precisely, the shape of reference to Z is sought in the "compactified" family

$$\mathbb{X}(Z) \stackrel{def}{=} \{X \in \mathbb{X}, \mathrm{Dig}_h(X \setminus \partial X) \subset Z \text{ and } \mathrm{Dig}_h(\mathbb{R}^d \setminus X) \subset \mathbb{Z}^d \setminus Z\}.$$

It just means that points exactly on the shape boundary may be either digitized "in" or "out". The shape of reference $X^{\mathrm{ref}}(Z)$ is the solution to the following minimization problem

$$X^{\mathrm{ref}}(Z) \stackrel{def}{=} \operatorname*{arg\,min}_{X \in \mathbb{X}(Z)} \int_{\partial X} \kappa^2 ds,$$

where κ is the mean curvature field over ∂X.

Definition 4. Let Z be a digital shape. Let p be any point inside some $d-1$-cell of $\Delta(Z)$. The *parameter-free MK2 curvature estimator* $\hat{\kappa}^\star_{\mathrm{MK}^2}$ is defined as

$$(4.1) \qquad\qquad \hat{\kappa}^\star_{\mathrm{MK}^2}(Z, p) \stackrel{def}{=} \kappa(X^{\mathrm{ref}}(Z), \xi_{\partial X^{\mathrm{ref}}(Z)}(p)),$$

where $\xi_{\partial X^{\mathrm{ref}}(Z)}$ is the closest point to p on the boundary of the shape of reference to Z.

The *rescaled MK2 curvature estimator* $\hat{\kappa}_{\mathrm{MK}^2}$ is naturally defined for some point $\hat{x} \in \partial_h X$ as

$$(4.2) \qquad\qquad \hat{\kappa}_{\mathrm{MK}^2}(\mathrm{Dig}_h(X), \hat{x}, h) \stackrel{def}{=} \frac{1}{h} \hat{\kappa}^\star_{\mathrm{MK}^2}(\mathrm{Dig}_h(X), \frac{1}{h}\hat{x}).$$

Finding this *shape of reference* is not a trivial task. In [20], a fast algorithm called *GMC* and using digital straight segments provides an approximation with no theoretical guarantees. This variational problem is also known as the minimization of a Willmore energy under constraints.

Hence, in [4] two other numerical techniques were proposed to find a solution: (i) a precise one based on convex optimization (but limited to convex shapes): (ii) a more versatile technique based on phase field approximation which is also extensible to 3D.

It is rather clear by definition that such an estimator should be uniformly multigrid convergent, provided one may determine the exact shape of reference. However, whatever the chosen algorithm or numerical technique, there is yet no theoretical guarantee on this estimator. A comprehensive 2D evaluation shows that it is experimentally multigrid convergent, although it is slightly less accurate than the MDCA estimator on perfect data. However, it is very stable and presents no oscillations in the result. It is thus easy to find the dominant points (maxima and minima of curvatures) and inflexion zones [21]. Another advantage of this approach is that it reconstructs a shape of reference. We have thus more than just an estimation of the curvature field. Figure 4.1 illustrates MK2 curvature estimations and contour reconstruction for 2D digital shapes, while Figure 4.2 gives 3D reconstruction results on a digital rabbit.

5. CURVATURE BY DIGITAL INTEGRAL INVARIANTS

Integral invariants were proposed in [35, 34] as a tool to analyze locally the geometry of triangulated mesh. The idea is to define integral quantities over the intersection of the shape X with a ball $B_r(x)$, centered on the point of interest x and of given radius r (see Fig. 5.1). These integral quantities are thus functions of the parameter r. For instance, the mean curvature is related to the volume of $X \cap B_r(x)$: it participates in the second term of the Taylor expansion of the volume at $r = 0$. We may note that a very similar tool was proposed earlier in [6].

It is possible to adapt this approach to digital data. In [7], the authors define a curvature estimator for 2D shapes and a mean curvature estimator for 3D shapes, based on digital integral invariants. The full curvature tensor is estimated by means of digital integral invariants in [8]. The 2D *parameter-free* curvature estimator is presented in [28].

Given a digital shape $Z \subset \mathbb{Z}^d$, the *discrete volume at step h* is defined as $\widehat{\mathrm{Vol}}_d(Z,h) \stackrel{def}{=} h^d \mathrm{Card}(Z)$.

Definition 5. Given a digital shape $Z \subset \mathbb{Z}^2$, any point $x \in \mathbb{R}^2$, some radius $r > 0$ and a gridstep $0 < h < r$, the *II curvature estimator* is defined as:

$$(5.1) \qquad \hat{\kappa}_{r,\mathrm{II}}(Z,x,h) \stackrel{def}{=} \frac{3\pi}{2r} - \frac{3\widehat{\mathrm{Vol}}_2(B_{r/h}(\frac{1}{h}x) \cap Z, h)}{r^3}.$$

The 3D extension to this estimator, when $Z \subset \mathbb{Z}^3$ and $x \in \mathbb{R}^3$, is called the *II mean curvature estimator* and is written as:

$$(5.2) \qquad \hat{\kappa}_{r,\mathrm{II}}^{m}(Z,x,h) \stackrel{def}{=} \frac{8}{3r} - \frac{4\widehat{\mathrm{Vol}}_3(B_{r/h}(\frac{1}{h}x) \cap Z, h)}{\pi r^4}.$$

When one wishes to estimate the full curvature tensor (principal curvatures, principal directions), we must estimate the second order moments of $X \cap B_r(x)$, also known as covariance matrix. For integers i, j, k, the i,j,k-*discrete moment of Z at step h* is defined as $\hat{m}_{i,j,k}(Z,h) \stackrel{def}{=} h^{i+j+k} \sum_{(x,y,z) \in Z} x^i y^j z^k$. The *digital covariance matrix* is naturally defined as a centered version of the tensor of second order discrete moments:

$$\widehat{J}(Z,h) \stackrel{def}{=} \left[\begin{array}{ccc} \hat{m}_{2,0,0}(Z,h) & \hat{m}_{1,1,0}(Z,h) & \hat{m}_{1,0,1}(Z,h) \\ \hat{m}_{1,1,0}(Z,h) & \hat{m}_{0,2,0}(Z,h) & \hat{m}_{0,1,1}(Z,h) \\ \hat{m}_{1,0,1}(Z,h) & \hat{m}_{0,1,1}(Z,h) & \hat{m}_{0,0,2}(Z,h) \end{array} \right]$$

$$- \frac{1}{\hat{m}_{0,0,0}(Z,h)} \left[\begin{array}{c} \hat{m}_{1,0,0}(Z,h) \\ \hat{m}_{0,1,0}(Z,h) \\ \hat{m}_{0,0,1}(Z,h) \end{array} \right] \otimes \left[\begin{array}{c} \hat{m}_{1,0,0}(Z,h) \\ \hat{m}_{0,1,0}(Z,h) \\ \hat{m}_{0,0,1}(Z,h) \end{array} \right]^{T}.$$

Following the truncated Taylor expansion of [35], Theorem 2, we define estimators of curvatures from the diagonalization of the digital covariance matrix. Note that principal direction estimators are simply the two main eigenvectors of this matrix.
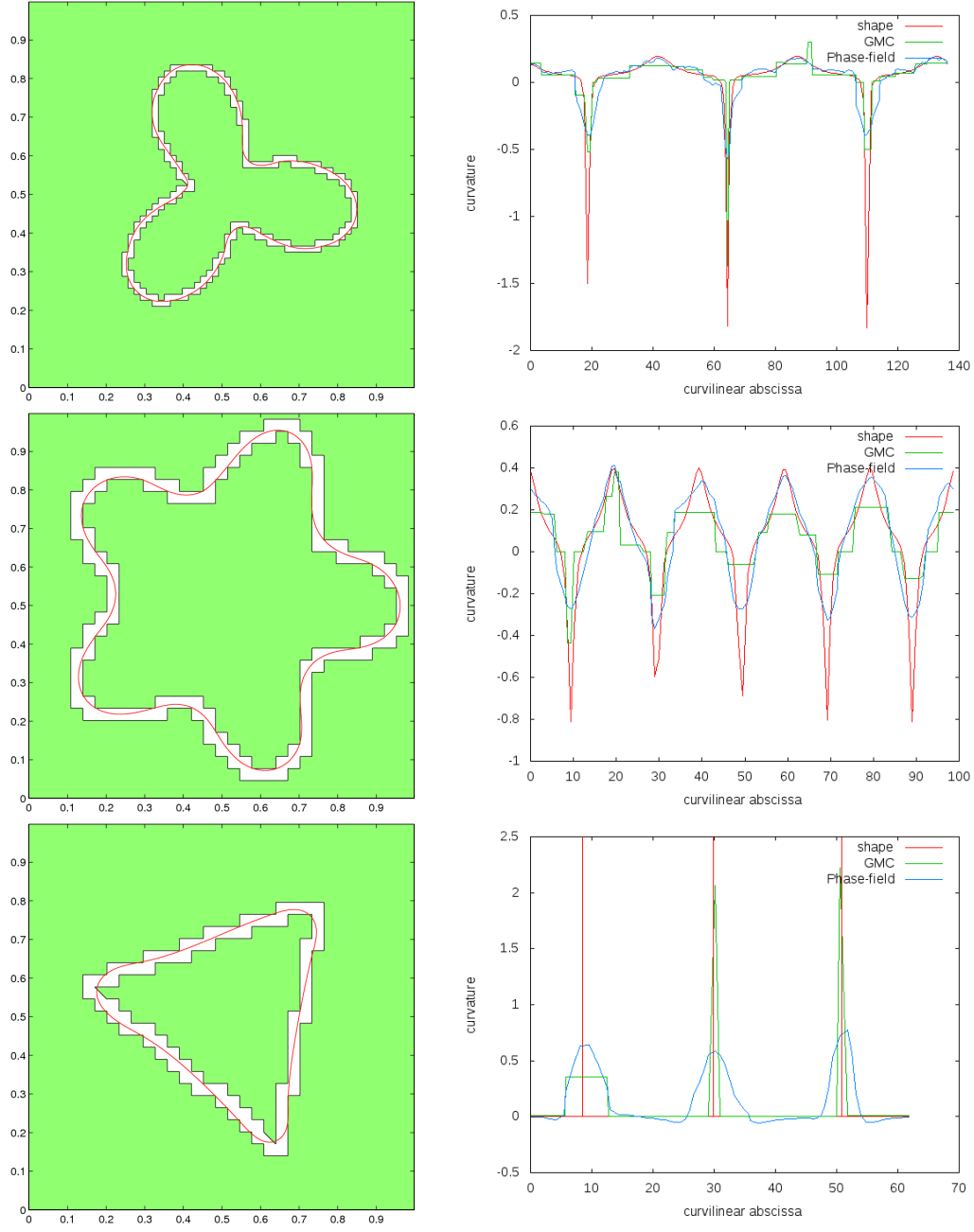
Figure 4.1: Digital shapes (left: contour as white digital path between interior and exterior pixels), shape of reference obtained with phase field reconstruction (left: red curve) and comparison of curvature estimations (right) with: true curvature (red), GMC algorithm (green) and Phase-field technique (cyan).

Definition 6. Given a digital shape $Z \subset \mathbb{Z}^3$, any point $x \in \mathbb{R}^3$, some radius $r > 0$ and a gridstep $0 < h < r$, the *II principal curvature estimators* are defined as:

$$(5.3) \qquad \hat{\kappa}_{r,\text{II}}^1(Z, x, h) \quad = \quad \frac{6}{\pi r^6}(\hat{\lambda}_2 - 3\hat{\lambda}_1) + \frac{8}{5r},$$

$$(5.4) \qquad \hat{\kappa}_{r,\text{II}}^2(Z, x, h) \quad = \quad \frac{6}{\pi r^6}(\hat{\lambda}_1 - 3\hat{\lambda}_2) + \frac{8}{5r},$$

where $\hat{\lambda}_1$ and $\hat{\lambda}_2$ are the two greatest eigenvalues of $\widehat{J}(B_{r/h}(\frac{1}{h}x) \cap Z, h)$.

<div align="center">

iteration 0   iteration 48

</div>

Figure 4.2: Phase field reconstruction of 3D digital rabbit.

<div align="center">

(a)   (b)

</div>

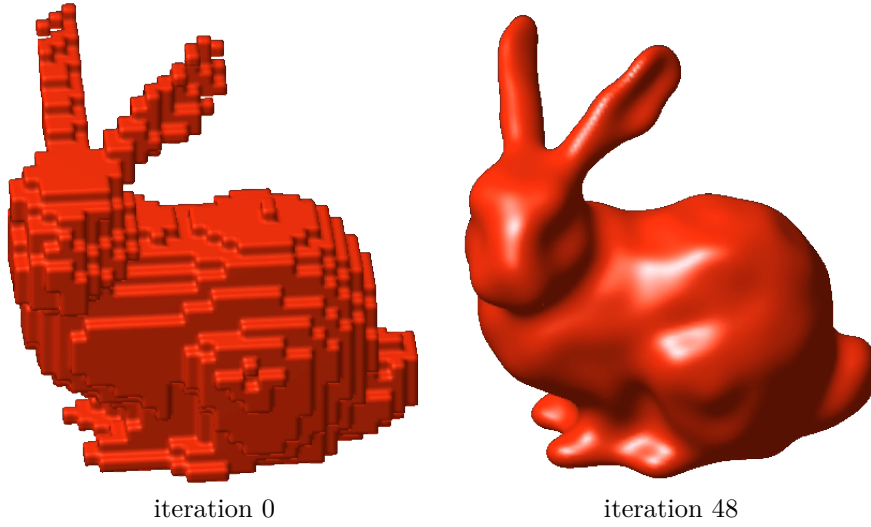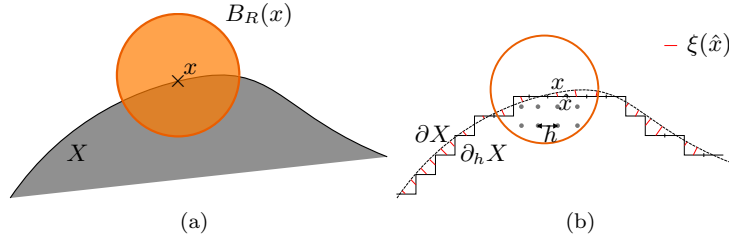Figure 5.1: Integral invariant computation (*left*) and notations (*right*) in dimension 2.

The II curvature estimator can be made parameter-free. The idea is to use the average discrete length of all *maximal segments* of $\Delta(Z)$. Any part C' of a digital contour C is a *digital straight segment* (DSS for short) if and only if the interior and exterior pixels of C' are linearly separable, i.e. there exists a Euclidean straight line that separates interior points from exterior points. Any DSS M of C is a *maximal segment* if and only if all the parts M' containing M, i.e. such that $M \subsetneq M' \subset C$, are not a DSS.

When the digital shape is the digitization of some Euclidean shape X at gridstep h, the discrete length of maximal segments follows several asymptotic relations [12]. If we denote by $L_D(Z)$ the average of the discrete length of all maximal segments on the contour $\Delta(Z)$, then the precise fact used here is

$$\Theta(h^{-\frac{1}{3}}) \leq L_D(\mathrm{Dig}_h(X)) \leq \Theta(h^{-\frac{1}{3}}\log(h^{-1})).$$

Definition 7. Let $Z \subset \mathbb{Z}^2$ be a digital shape, and $C = \Delta(Z)$ its digital contour. Let p be any point of a linel of C. The *parameter-free II curvature estimator* $\hat{\kappa}_{II}^{\star}$ is defined as:

$$\hat{\kappa}_{II}^{\star}(Z,p) \overset{def}{=} \frac{3\pi}{2\rho(Z)} - \frac{3A(Z,p)}{\rho(Z)^3} \tag{5.5}$$

where $\rho(Z) = (L_D(Z))^2$ and $A(Z,p) = \mathrm{Card}(B_{\rho(Z)}(p) \cap Z)$.

The *rescaled II curvature estimator* $\hat{\kappa}_{II}$ is naturally defined for some point $\hat{x} \in \partial_h X$ as

$$\hat{\kappa}_{II}(\mathrm{Dig}_h(X), \hat{x}, h) \overset{def}{=} \frac{1}{h}\hat{\kappa}_{II}^{\star}(\mathrm{Dig}_h(X), \frac{1}{h}\hat{x}). \tag{5.6}$$

We do have several multigrid convergence for these estimators, for the family of shapes $\mathbb{X}$ (compact sets, C^3-smooth boundary):
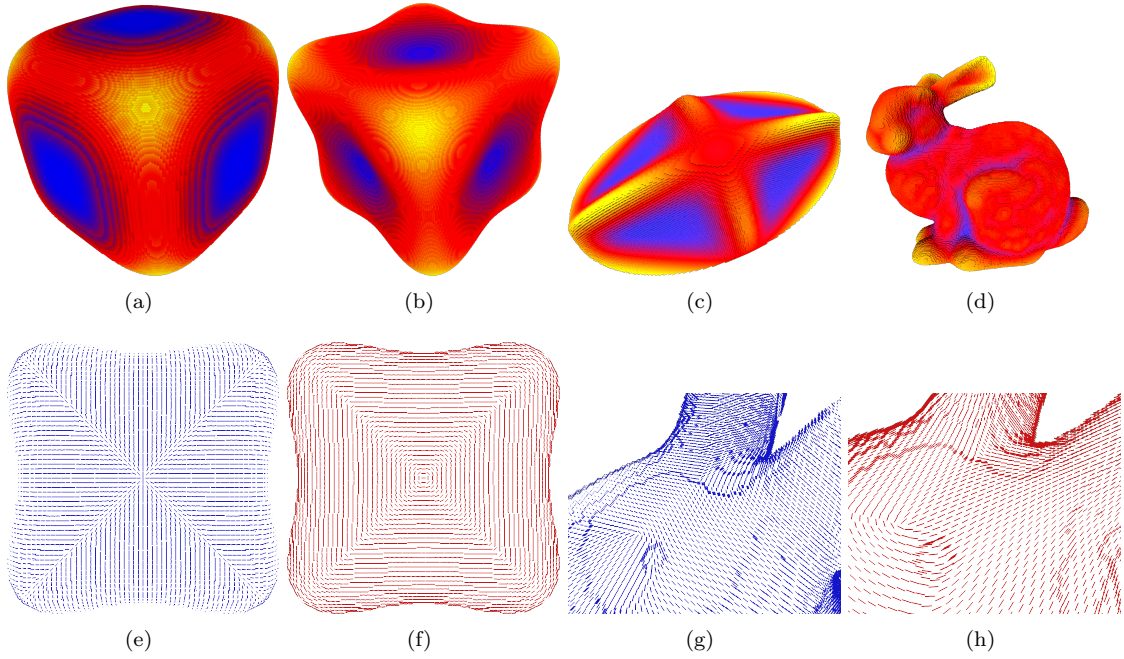
Figure 5.2: Illustration of 3D curvature estimation. Mean curvature on rounded cube (*a*), Goursat's surface (*b*), Leopold surface (*c*) and a bunny (*d*). First principal direction and second principal direction Goursat's surface (*e* and *f*) and Stanford bunny (*g* and *h*) .

estimator	quantity	parameters	convergence speed	reference
$\hat{\kappa}_{r,\mathrm{II}}$	2D curvature	$r = h^{\frac{1}{3}}$	$O(h^{\frac{1}{3}})$	[7]
$\hat{\kappa}_{r,\mathrm{II}}^{m}$	3D mean curvature	$r = h^{\frac{1}{3}}$	$O(h^{\frac{1}{3}})$	[7]
$\hat{\kappa}_{r,\mathrm{II}}^{1}$	1st principal curvature	$r = h^{\frac{1}{3}}$	$O(h^{\frac{1}{3}})$	[8]
$\hat{\kappa}_{r,\mathrm{II}}^{2}$	2nd principal curvature	$r = h^{\frac{1}{3}}$	$O(h^{\frac{1}{3}})$	[8]
$\hat{\kappa}_{\mathrm{II}}^{\star}$	2D unscaled curvature	parameter-free	$O(h^{\frac{1}{3}} \log^2(h^{-1}))$	[28]
$\hat{\kappa}_{\mathrm{II}}$	2D curvature	scale h	$O(h^{\frac{1}{3}} \log^2(h^{-1}))$	[28]

A very comprehensive set of experimental evaluation has been performed on II curvature estimators [7, 8, 28], as well as many comparisons with other approaches (MDCA, binomial convolutions [29, 15], jet fitting [5]). It is of course experimentally multigrid convergent. It is one of the most accurate in practice. Furthermore it is robust to noise due to its integral form. Figure 5.2 displays some results of estimators $\hat{\kappa}_{r,\mathrm{II}}^{m}$ and directions of $\hat{\kappa}_{r,\mathrm{II}}^{1}$ and $\hat{\kappa}_{r,\mathrm{II}}^{2}$. This has been implemented in the open-source library DGtal [14].

6. Discussion

We have presented three families of digital curvature estimators. It is clear that the gridstep h is necessary to get the correct *unit* for curvature, but we have shown above that we can define curvature estimators requiring *no parameter* if we assume simply that a pixel or voxel has unit length. Even better, we have exhibited one curvature estimator, the parameter-free II curvature estimator $\hat{\kappa}_{\mathrm{II}}^{\star}$, the multigrid convergence of which is established for the classical family $\mathbb{X}$ of compact shapes with C^3-boundary. Experiments show that this estimator competes with the accurate MDCA estimator but with the advantage of theoretical guarantees as well as a robustness to noise.

The following table summarizes the respective qualities of each curvature estimator, according to the desired properties described in the introduction. When an estimator meets fully a property, it is circled by a frame. The symbol n stands for the number of elements of $\Delta(Z)$. Note that $n = \Theta(h^{d-1})$ if d is the dimension of the space.

	convergence	accuracy	exact comp.	efficient	robust	parameters
2D estimators						
$\hat{\kappa}^{\star}_{\text{MDCA}}$	?	$O(h^{\frac{1}{3}})$	yes	$O(n^{\frac{4}{3}})$	no	unit
$\hat{\kappa}^{\star}_{\text{MK}^2}$ (GMC)	?	$\approx O(h^{\frac{1}{3}})$	Opt.	iter $\times O(n^{\frac{2}{3}})$	Hausdorff	$\approx$ unit
$\hat{\kappa}^{\star}_{\text{MK}^2}$ (PF)	?	$O(h^{\frac{1}{3}})$	Opt.	iter $\times O(n^2)$	no	$\approx$ unit
$\hat{\kappa}_{r,\text{II}}$	$O(h^{\frac{1}{3}})$	$O(h^{\frac{1}{3}})$	Yes	$O(n^{\frac{5}{3}})$	yes	need h
$\hat{\kappa}^{\star}_{\text{II}}$	$O(h^{\frac{1}{3}})$	$O(h^{\frac{1}{3}})$	Yes	$O(n^{\frac{5}{3}})$	yes	unit
3D estimators						
$\hat{\kappa}^{\star}_{\text{MK}^2}$ (PF)	?	?	Opt.	iter $\times O(n^{\frac{3}{2}})$	no	$\approx$ unit
$\hat{\kappa}^{m}_{r,\text{II}}$	$O(h^{\frac{1}{3}})$	$O(h^{\frac{1}{3}})$	Yes	$O(n^{\frac{5}{3}})$	yes	need h
$\hat{\kappa}^{1}_{r,\text{II}}$	$O(h^{\frac{1}{3}})$	$O(h^{\frac{1}{3}})$	Yes	$O(n^{\frac{5}{3}})$	yes	need h
$\hat{\kappa}^{2}_{r,\text{II}}$	$O(h^{\frac{1}{3}})$	$O(h^{\frac{1}{3}})$	Yes	$O(n^{\frac{5}{3}})$	yes	need h

It is clear that the next step is to define parameter-free 3D principal curvature estimators, with guaranteed multigrid convergence. For now, for this problem, only empirical solutions exist.

Relevance of digital estimators for estimating curvatures in gray-level images. Another natural question is the suitability of using digital curvature estimators on 2D or 3D gray-level image data. In this case, the input data is much richer than a simple binary image, since grey-level values could potentially be used for determining curvatures. Therefore there exists standard image derivation techniques to estimate the curvature of isocontours or isosurfaces within images, some involve derivative filters (e.g. [32]), specialized finite difference schemes [39, 33], or image structure tensor [37, 25].

Since it is parameter-free, we examine here the upwind finite difference scheme used in Level-Set (LS) techniques for estimating the mean curvature of some isosurface [39, 33], and we compare its accuracy to the 3D Integral Invariant (II) mean curvature estimator. Given a point p of value $I(p)$ in image I, LS curvature estimator uses grey-level information around p and estimates the mean curvature at p of the isosurface of value $I(p)$. On the other hand, II estimator processes only the **binary** image obtained by thresholding I at the value $I(p)$. We have compared numerically the respective performance of LS and II on a 3D image of a ball of radius 30, with a linear gradient of 50 at its boundary. We have also add a Gaussian noise of standard deviation $\sigma \in \{0, 1, 2, 3, 4, 5\}$, which is a very small perturbation considering that the background is 50 and the foreground is 200. Input data and experiments are illustrated on Figure 6.1. As one can see, even the noise with deviation 5 is almost imperceptible. If we look however at the average mean curvature computed by LS and II estimators, we see that their behaviors are dramatically different. LS estimator is accurate if the image is perfect (average is very close with 0.8% relative error but samples have a relative deviation of 38%). However, as soon a slight perturbation is added to the data, this estimator becomes very unstable. On the other hand, the accuracy of II estimator is related to a correct choice of ball radius (here $r \approx 10$ gives excellent results), but this estimator is stable whatever the noise. Note that the discussion above gives indication for the correct radius. Indeed it is as if we are digitizing a ball of radius 1 with gridstep $h = \frac{1}{30}$. A correct Euclidean radius for II estimator should follow $h^{\frac{1}{3}}$, hence the corresponding discrete radius is $\frac{h^{\frac{1}{3}}}{h} = h^{-\frac{2}{3}} \approx 9.65$.

To conclude, the image curvature estimator of [39, 33] is too unstable for analyzing real images coming from camera or biomedical devices. More robust techniques using image structure tensor [37, 25] can be parameterized to address noise in a global manner. The II digital curvature estimator gives reasonably accurate results even if the gray-level information is not reliable, and is stable with respect to noise (with zero mean). A natural open question is to extend digital curvature estimator to gray-level images (hence the shape is defined as a fuzzy characteristic set). II estimator may be a good candidate since the covariance matrix can be weighted accordingly.
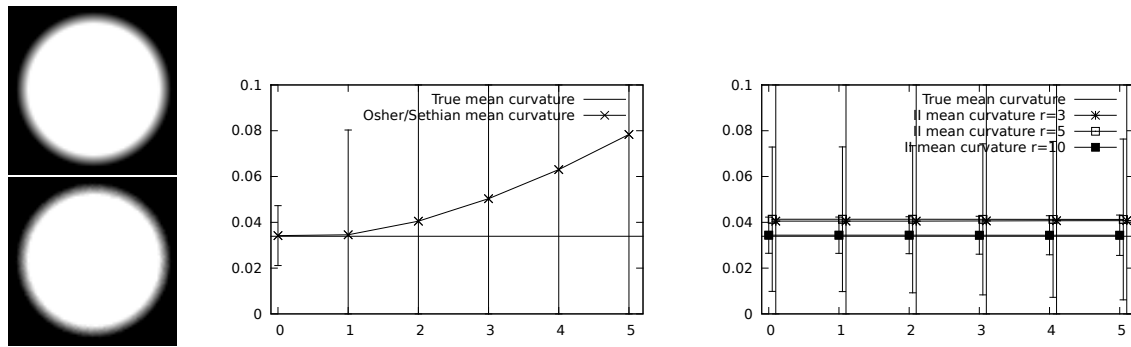
Figure 6.1: Mean curvature computation in a 3D gray-level image. **Left:** Slices in the input 3D 8-bit gray-level image, which represent a 3D ball of radius 30 with gradient 50 around its isosurface 128 (top image is perfect data, bottom image is data damaged with a Gaussian noise of deviation 5, i.e. PSNR=84.5). **Middle:** Average and deviation of mean curvature computed with LS estimator. **Right:** Average and deviation of mean curvature computed with II estimator, with several radii.

References

[1] P. Alliez, D. Cohen-Steiner, Y. Tong, and M. Desbrun. Voronoi-based variational reconstruction of unoriented point sets. In *Symposium on Geometry processing*, volume 7, pages 39–48, 2007.

[2] N. Amenta, M. Bern, and M. Kamvysselis. A new voronoi-based surface reconstruction algorithm. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 415–421. ACM, 1998.

[3] A. I. Bobenko and Y. B. Suris. *Discrete differential geometry: Integrable structure*, volume 98. AMS Bookstore, 2008.

[4] E. Bretin, J.-O. Lachaud, and É. Oudet. Regularization of discrete contour by willmore energy. *Journal of Mathematical Imaging and Vision*, 40:214–229, 2011.

[5] F. Cazals and M. Pouget. Estimating differential quantities using polynomial fitting of osculating jets. *Computer Aided Geometric Design*, 22(2):121–146, 2005.

[6] U. Clarenz, M. Rumpf, and A. Telea. Robust feature detection and local classification for surfaces based on moment analysis. *Visualization and Computer Graphics, IEEE Transactions on*, 10(5):516–524, 2004.

[7] D. Coeurjolly, J.-O. Lachaud, and J. Levallois. Integral based curvature estimators in digital geometry. In *Discrete Geometry for Computer Imagery*, number 7749 in LNCS, pages 215–227. Springer, 2013.

[8] D. Coeurjolly, J.-O. Lachaud, and J. Levallois. Multigrid convergent principal curvature estimators in digital geometry. *Computer Vision and Image Understanding*, 2014. Submitted, minor revision.

[9] D. Coeurjolly, J.-O. Lachaud, and T. Roussillon. Multigrid convergence of discrete geometric estimators. In V. Brimkov and R. Barneva, editors, *Digital Geometry Algorithms, Theoretical Foundations and Applications of Computational Imaging*, volume 2 of *Lecture Notes in Computational Vision and Biomechanics*, pages 395–424. Springer, 2012.

[10] D. Cohen-Steiner and J.-M. Morvan. Restricted delaunay triangulations and normal cycle. In *Proceedings of the nineteenth annual symposium on Computational geometry*, SCG'03, pages 312–321, New York, NY, USA, 2003. ACM.

[11] D. Cohen-Steiner and J.-M. Morvan. Second fundamental measure of geometric sets and local approximation of curvatures. *Journal of Differential Geometry*, 74(3):363–394, 2006.

[12] F. de Vieilleville, J.-O. Lachaud, and F. Feschet. Maximal digital straight segments and convergence of discrete geometric estimators. *Journal of Mathematical Image and Vision*, 27(2):471–502, 2007.

[13] M. Desbrun, A. N. Hirani, M. Leok, and J. E. Marsden. Discrete exterior calculus. *arXiv preprint math/0508341*, 2005.

[14] DGtal: Digital geometry tools and algorithms library. `http://libdgtal.org`.

[15] H.-A. Esbelin, R. Malgouyres, and C. Cartade. Convergence of binomial-based derivative estimation for 2 noisy discretized curves. *Theoretical Computer Science*, 412(36):4805 – 4813, 2011.

[16] H. Federer. Curvature measures. *Trans. Amer. Math. Soc*, 93(3):418–491, 1959.

[17] S. Fourey and R. Malgouyres. Normals and curvature estimation for digital surfaces based on convolutions. In *Discrete Geometry for Computer Imagery*, LNCS, pages 287–298. Springer, 2008.

[18] T. D. Gatzke and C. M. Grimm. Estimating curvature on triangular meshes. *International Journal of Shape Modeling*, 12(01):1–28, 2006.

[19] B. Kerautret and J.-O. Lachaud. Robust estimation of curvature along digital contours with global optimization. In D. Coeurjolly, I. Sivignon, L. Tougne, and F. Dupont, editors, *Proc. Int. Conf. Discrete Geometry for Computer Imagery (DGCI'2008), Lyon, France*, volume 4992 of *LNCS*, pages 334–345. Springer, April 2008.

[20] B. Kerautret and J.-O. Lachaud. Curvature estimation along noisy digital contours by approximate global optimization. *Pattern Recognition*, 42(10):2265 – 2278, 2009.

[21] B. Kerautret, J.-O. Lachaud, and B. Naegel. Curvature based corner detector for discrete, noisy and multi-scale contours. *International Journal of Shape Modeling*, 14(2):127–145, 2008.

[22] R. Klette and A. Rosenfeld. *Digital Geometry: Geometric Methods for Digital Picture Analysis*. Series in Computer Graphics and Geometric Modelin. Morgan Kaufmann, 2004.

[23] R. Klette and J. Žunić. Multigrid convergence of calculated features in image analysis. *Journal of Mathematical Imaging and Vision*, 13(3):173–191, 2000.

[24] J.-O. Lachaud. *Espaces non-euclidiens et analyse d'image : modèles déformables riemanniens et discrets, topologie et géométrie discrète*. Habilitation à diriger des recherches, Université Bordeaux 1, Talence, France, 2006.

[25] J-O Lachaud and Benjamin Taton. Deformable model with a complexity independent from image resolution. *Computer Vision and Image Understanding*, 99(3):453–475, 2005.

[26] J.-O. Lachaud, A. Vialard, and F. de Vieilleville. Fast, accurate and convergent tangent estimation on digital contours. *Image and Vision Computing*, 25(10):1572–1587, October 2007.

[27] A. Lenoir. Fast estimation of mean curvature on the surface of a 3d discrete object. In E. Ahronovitz and C. Fiorio, editors, *Proc. Discrete Geometry for Computer Imagery (DGCI'97)*, volume 1347 of *Lecture Notes in Computer Science*, pages 175–186. Springer Berlin Heidelberg, 1997.

[28] J. Levallois, D. Coeurjolly, and J.-O. Lachaud. Parameter-free and multigrid convergent digital curvature estimators. In *Discrete Geometry for Computer Imagery*, 2014. Submitted.

[29] R. Malgouyres, F. Brunet, and S. Fourey. Binomial convolutions and derivatives estimation from noisy discretizations. In *Discrete Geometry for Computer Imagery*, volume 4992 of *LNCS*, pages 370–379. Springer, 2008.

[30] Q. Mérigot, M. Ovsjanikov, and L. Guibas. Robust voronoi-based curvature and feature estimation. In *2009 SIAM/ACM Joint Conference on Geometric and Physical Modeling*, SPM'09, pages 1–12, New York, NY, USA, 2009. ACM.

[31] Q. Mérigot, M. Ovsjanikov, and L. Guibas. Voronoi-based curvature and feature estimation from point clouds. *Visualization and Computer Graphics, IEEE Transactions on*, 17(6):743–756, 2011.

[32] O. Monga and S. Benayoun. Using partial derivatives of 3d images to extract typical surface features. *Computer vision and image understanding*, 61(2):171–189, 1995.

[33] S. Osher and N. Paragios. *Geometric level set methods in imaging, vision, and graphics*. Springer, 2003.

[34] H. Pottmann, J. Wallner, Q. Huang, and Y. Yang. Integral invariants for robust geometry processing. *Computer Aided Geometric Design*, 26(1):37–60, 2009.

[35] H. Pottmann, J. Wallner, Y. Yang, Y. Lai, and S. Hu. Principal curvatures from the integral invariant viewpoint. *Computer Aided Geometric Design*, 24(8-9):428–442, 2007.

[36] L. Provot and Y. Gérard. Estimation of the derivatives of a digital function with a convergent bounded error. In *Discrete Geometry for Computer Imagery*, LNCS, pages 284–295. Springer, 2011.

[37] B. Rieger, F. J. Timmermans, L. J. Van Vliet, and P. W. Verbeek. On curvature estimation of iso surfaces in 3d gray-value images and the computation of shape descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(8):1088–1094, 2004.

[38] T. Roussillon and J.-O. Lachaud. Accurate curvature estimation along digital contours with maximal digital circular arcs. In *Combinatorial Image Analysis*, volume 6636, pages 43–55. Springer, 2011.

[39] James Albert Sethian. *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, volume 3. Cambridge university press, 1999.

[40] F. Sloboda and J. Stoer. On piecewise linear approximation of planar Jordan curves. *J. Comput. Appl. Math.*, 55(3):369–383, 1994.

[41] T. Surazhsky, E. Magid, O. Soldea, G. Elber, and E. Rivlin. A comparison of gaussian and mean curvatures estimation methods on triangular meshes. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 1, pages 1021–1026, 2003.

[42] G. Xu. Convergence analysis of a discretization scheme for gaussian curvature over triangular surfaces. *Computer Aided Geometric Design*, 23(2):193–207, 2006.

Univ. Savoie, LAMA, F-73000 Chambéry, France — CNRS, LAMA, F-73000 Chambéry, France • jacques-olivier.lachaud@univ-savoie.fr

Curvature and Flow in Digital Space

Atsushi Imiya

Abstract

We first define the curvature indices of vertices of digital objects. Second, using these indices, we define the principal normal vectors of digital curves and surfaces. These definitions allow us to derive the Gauss-Bonnet theorem for digital objects. Third, we introduce curvature flow for isothetic polytopes defined in a digital space.

1. Introduction

A unified treatment of shape deformation is required for the intelligent editing of image contents for multimedia technology. The deformation of image data based on curvature flow and diffusion processes [17, 18, 29, 21] on surfaces [13, 22] provides a mathematical foundation for the unified treatment of deformation [23].

These deformation operations for boundaries are discussed in the framework of the free boundary problem in mathematics. For the construction of solutions of partial differential equations as deformed surfaces, the numerical computation is achieved using an appropriate discretization scheme [13, 1]. Bruckstein *et al.* derived a digital version of this problem for planar shapes [7]. Furthermore, Bobenko and Suris proposed a spatial version of their digital treatment [6].

In this paper, we introduce a transform [12] for a binary digital set [14], which we call *Digital Curvature Flow*. Digital curvature flow describes the geometric flow [7, 10] controlled by the curvature on the boundary of binary digital images on a plane and in a space. This flow, which moves the boundary, can also be considered as the curvature flow on isothetic polytopes of which all edges are parallel to axes of the orthogonal coordinate system [14].

For the numerical analysis of partial differential equations, it is required to generate grids [28] or decompose the region of interest to small domains [27] for the discretization of equations [19, 9, 20]. Therefore, numerical analysis is achieved in discrete forms. However, these grids usually depend on problems which we want to deal with. In contrast to this classical numerical treatment [27, 28], in this paper, we define the digital treatment of the deformation of the boundary of an object in a digital space which is defined as a collection of lattice points [14].

Control of the topology is an important problem for flow-based shape analysis and processing [13, 22, 7]. Curvature flows usually cause the collapse of the topology. Therefore, we propose a method for the examination of the topology of digital shapes and objects. Using this process we can detect collapses of topologies and control the topology of shapes and objects during deformation by flow-based processing.

Since the curvature of a point on curves and surfaces is defined locally [26, 24, 31, 25], the curvature flow is basically a local operation on them. As a digital treatment of partial differential equations, the theory of cellular automaton is proposed. In this theory, a space is also digitized and equations are approximately expressed as rules which rewrite the configurations of 1-points in a neighbourhood of a digital space [15, 14]. Therefore, rules for the cellular-automaton treatment [16, 8, 30] of partial differential equations are basically local operations. These similar properties

between cellular automata and curvature suggest that our treatment of boundary deformation is natural from the viewpoint of space discretization.

2. Connectivity and Neighbourhood

Setting $\mathbf{R}^2$ and $\mathbf{R}^3$ to be two- and three-dimensional Euclidean spaces, we express vectors in $\mathbf{R}^2$ and $\mathbf{R}^3$ as $\boldsymbol{x} = (x, y)^\top$ and $\boldsymbol{x} = (x, y, z)^\top$, respectively, where $\top$ is the transpose of the vector. Setting $\mathbf{Z}$ to be the set of all integers, the two- and three-dimensional digital spaces $\mathbf{Z}^2$ and $\mathbf{Z}^3$ are sets of points such that both x and y are integers and x, y and z are all integers, respectively.

On $\mathbf{Z}^2$ and in $\mathbf{Z}^3$

$$(2.1) \qquad \mathbf{N}^4((m, n)^\top) = \{(m \pm 1, n)^\top, (m, n \pm 1)^\top\}$$

and

$$\mathbf{N}^6((k, m, n)^\top) = \{(k \pm 1, m, n)^\top, (k, m \pm 1, n)^\top, (k, m, n \pm 1)^\top\}$$

are the planar 4-neighbourhood of point $(m, n)^\top$ and the spatial 6-neighbourhood of point $(k, m, n)^\top$, respectively. In this paper, we assume 4-connectivity on $\mathbf{Z}^2$ and 6-connectivity in $\mathbf{Z}^3$.

For integers k, m and n, the collection of integer triplets (k', m', n') which satisfy the equation

$$(2.2) \qquad (k - k')^2 + (m - m')^2 + (n - n')^2 = 1$$

defines points in the 6-neighbourhood of point $(k, m, n)^\top$. If we substitute $k = k'$, $m = m'$ and $n = n'$ into eq. (2.2), we obtain the equations

$$(2.3) \qquad (m - m')^2 + (n - n')^2 = 1, \ \ (k - k')^2 + (n - n')^2 = 1, \ \ (m - m')^2 + (n - n')^2 = 1.$$

These equations define points in the planar 4-neighbourhoods. Therefore, setting one of x, y or z to be a fixed integer, we obtain two-dimensional sets of lattice points such that

$$(2.4) \qquad \mathbf{Z}_1^2((k, m, n)^\top) = \{(k, m, n)^\top | \exists k, \forall m, \forall n \in \mathbf{Z}\},$$

$$(2.5) \qquad \mathbf{Z}_2^2((k, m, n)^\top) = \{(k, m, n)^\top | \forall k, \exists m, \forall n \in \mathbf{Z}\},$$

and

$$(2.6) \qquad \mathbf{Z}_3^2((k, m, n)^\top) = \{(k, m, n)^\top | \forall k, \forall m, \exists n \in \mathbf{Z}\}.$$

These two-dimensional digital spaces are mutually orthogonal. Denoting

$$(2.7) \qquad \mathbf{N}_1^4((k, m, n)^\top) = \{(k, m \pm 1, n)^\top, (k, m, n \pm 1)^\top\},$$

$$(2.8) \qquad \mathbf{N}_2^4((k, m, n)^\top) = \{(k \pm 1, m, n)^\top, (k, m, n \pm 1)^\top\},$$

and

$$(2.9) \qquad \mathbf{N}_3^4((k, m, n)^\top) = \{(k \pm 1, m, n)^\top, (k, m \pm 1, n)^\top\},$$

the relationship

$$\mathbf{N}^6((k, m, n)^\top) = \mathbf{N}_1^4((k, m, n)^\top) \cup \mathbf{N}_2^4((k, m, n)^\top) \cup \mathbf{N}_3^4((k, m, n)^\top)$$

holds since $\mathbf{N}_i^4((k, m, n)^\top)$ is the 4-neighbourhood on the plane $\mathbf{Z}_i^2((k, m, n)^\top)$ for $i = 1, 2, 3$ [11]. Equation (2.10) implies that the 6-neighbourhood is decomposed into three mutually orthogonal 4-neighbourhoods [11]. Figures 2.1 (a) and (b) illustrate the spatial 6-neighbourhood and planar 4-neighbourhood, respectively. The 6-neighbourfood in $\mathbf{Z}^3$ is decomposed into three mutually orthogonal 4-neighbourhoods as shown in Figure 2.1 (c).

A pair of points $(k, m, n)^\top$ and $\boldsymbol{x} \in \mathbf{N}^6((k, m, n)^\top)$ defines a unit line segment in $\mathbf{Z}^3$. Furthermore, four 6-connected points which form a circle define a unit plane segment in $\mathbf{Z}^3$ with respect to the 6-connectivity. Therefore, we assume that our object is a complex of $2 \times 2 \times 2$ cubes which share at least one face with each other. Thus, the surface of an object is a collection of unit squares which are parallel to the planes $x = 0$, $y = 0$ and $z = 0$.

Definition 1. For a point $\boldsymbol{x} \in \mathbf{Z}^3$, the collection of eight points

$$(2.10) \quad S^3 = \{\boldsymbol{x}, \boldsymbol{x} + \boldsymbol{e}_1, \boldsymbol{x} + \boldsymbol{e}_2, \boldsymbol{x} + \boldsymbol{e}_3, \boldsymbol{x} + \boldsymbol{e}_1 + \boldsymbol{e}_2, \boldsymbol{x} + \boldsymbol{e}_2 + \boldsymbol{e}_3, \boldsymbol{x} + \boldsymbol{e}_3 + \boldsymbol{e}_1, \boldsymbol{x} + \boldsymbol{e}_1 + \boldsymbol{e}_2 + \boldsymbol{e}_2\}$$

is a digital 3-simplex.

Definition 2. A digital object is a complex of a finite number of digital 3-simplices.
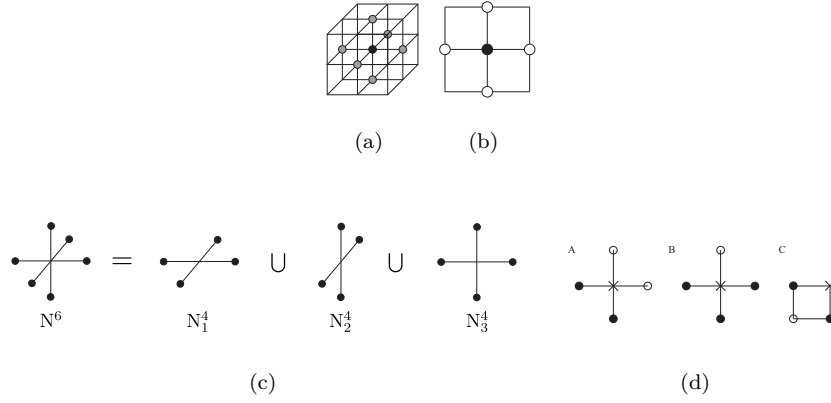
Figure 2.1: Local configuration of 2-manifold in $\mathbf{Z}^3$. (a) 6-neighbourhood in $\mathbf{Z}^3$ (b) 4-neighbourhood on $\mathbf{Z}^2$. (c) Decomposition of the 6-neighbourhood in the space to 4-neighbourhoods on the three orthogonal planes. (d) Configurations of points on the planar boundary. The spatial 6-neighbourfood in $\mathbf{Z}^3$ is decomposed into three mutually orthogonal 4-neighbourhoods.

3. Digital Boundary Manifold

For a pair of sets $\boldsymbol{A}$ and $\boldsymbol{B}$ in $\mathbf{R}^n$, the Minkowski addition $\boldsymbol{A} \oplus \boldsymbol{B}$ and the Minkowski subtraction $\boldsymbol{A} \ominus \boldsymbol{B}$ are defined as

$$(3.1) \qquad \boldsymbol{A} \oplus \boldsymbol{B} = \bigcup_{\boldsymbol{b} \in \boldsymbol{B}} \boldsymbol{A}(\boldsymbol{b}), \ \ \boldsymbol{A} \ominus \boldsymbol{B} = \bigcap_{\boldsymbol{b} \in \boldsymbol{B}} \boldsymbol{A}(\boldsymbol{b}),$$

for $\boldsymbol{A}(\boldsymbol{x}) = \{\boldsymbol{y} | \boldsymbol{y} = \boldsymbol{a} + \boldsymbol{x}, \forall \boldsymbol{a} \in \boldsymbol{A}\}$.

For the Minkowski addition and subtraction, the relations

$$(3.2) \qquad \boldsymbol{F} \ominus \boldsymbol{G} \ = \ \overline{\boldsymbol{F} \oplus \overline{\boldsymbol{G}}}$$

$$(3.3) \qquad \boldsymbol{F} \oplus (\boldsymbol{G} \cup \boldsymbol{H}) \ = \ (\boldsymbol{F} \oplus \boldsymbol{G}) \cup (\boldsymbol{F} \oplus \boldsymbol{H})$$

$$(3.4) \qquad \boldsymbol{F} \ominus (\boldsymbol{G} \cup \boldsymbol{H}) \ = \ (\boldsymbol{F} \ominus \boldsymbol{G}) \cap (\boldsymbol{F} \ominus \boldsymbol{H})$$

are satisfied. Furthermore, we obtain the following theorem.

Theorem 3. *If $\boldsymbol{F} \cap \boldsymbol{G} = \emptyset$, the equality*

$$(3.5) \qquad (\boldsymbol{F} \cup \boldsymbol{G}) \ominus \boldsymbol{H} = (\boldsymbol{F} \ominus \boldsymbol{H}) \cup (\boldsymbol{G} \ominus \boldsymbol{H})$$

is satisfied

(*Proof*)

$$\begin{aligned}
(\boldsymbol{F} \cup \boldsymbol{G}) \ominus \boldsymbol{H} \ &= \ \bigcap_{\boldsymbol{x} \in \boldsymbol{H}} (\boldsymbol{F} \cup \boldsymbol{G})(\boldsymbol{x}) \\
&= \ \{\boldsymbol{x} + \boldsymbol{y} | \forall \boldsymbol{x} \in \boldsymbol{H}, \exists \boldsymbol{y} \in (\boldsymbol{F} \cup \boldsymbol{G})\} \\
&= \ \{\boldsymbol{x} + \boldsymbol{y} | \forall \boldsymbol{x} \in \boldsymbol{H}, \exists \boldsymbol{y} \in \boldsymbol{F}\} \cup \{\boldsymbol{x} + \boldsymbol{y} | \forall \boldsymbol{x} \in \boldsymbol{H}, \exists \boldsymbol{y} \in \boldsymbol{G}\} \\
&= \ (\boldsymbol{F} \ominus \boldsymbol{H}) \cup (\boldsymbol{G} \ominus \boldsymbol{H})
\end{aligned}$$

$\square$

In $\mathbf{Z}^3$, the inner boundary of $\mathbf{F}$ is defined as

$$(3.6) \qquad \partial \mathbf{F} = \mathbf{F} \setminus (\mathbf{F} \ominus \mathbf{N}^6),$$

where $\mathbf{N}^6 = \mathbf{N}^6((0,0,0)^\top)$. A polyhedron whose vertices are points in $\partial \mathbf{F}$ is a Nef polyhedron [4, 2, 3, 5]. Therefore, the inner boundary of $\mathbf{F}$ is a Nef polyhedron.

Definition 4. We call a Nef polyhedron in $\mathbf{Z}^3$ a grid Nef polyhedron.

In this paper, we deal with the topological and geometrical properties of grid points on the surface of a grid Nef polyhedron.

For $i = 1, 2, 3$, we set

$$(3.7) \qquad \mathbf{Z}^2_{k(i)} = \{\boldsymbol{x} | \boldsymbol{x} = \boldsymbol{z} + k(i)\boldsymbol{e}_i, \boldsymbol{z} \in \mathbf{Z}^2_i((0,0,0)^\top)\},$$

where $\boldsymbol{e}_1 = (1,0,0)^\top$, $\boldsymbol{e}_2 = (0,1,0)^\top$ and $\boldsymbol{e}_3 = (0,0,1)^\top$. A slice of $\mathbf{F}$ perpendicular to the plane $\mathbf{Z}^2_i$ for $i = 1, 2, 3$ is

$$(3.8) \qquad \mathbf{F}^2_{k(i)} = \mathbf{F} \cap \mathbf{Z}^2_{k(i)},$$

Furthermore, we set

$$(3.9) \qquad K(i) = \{k(i) | \mathbf{F} \cap \mathbf{Z}^2_{k(i)} \neq \emptyset\},$$

that is, $K(i)$ is the number of slices which are perpendicular to $\boldsymbol{e}_i$. We have the following decomposition theorem.

Theorem 5. *Setting*

$$(3.10) \qquad \partial\mathbf{F}^2_{k(i)} = \mathbf{F}^2_{k(i)} \setminus (\mathbf{F}^2_{k(i)} \ominus \mathbf{N}^4_i),$$

where $\mathbf{N}^4_i = \mathbf{N}((0,0,0)^\top)$, the relation

$$(3.11) \qquad \partial\mathbf{F} = \bigcup_{i=1}^{3} \bigcup_{k(i) \in K(i)} \partial\mathbf{F}^2_{k(i)}$$

is satisfied.

(*Proof*) Since $\mathbf{N}^6 = \bigcup_{i=1}^{3} \mathbf{N}^4_i$, we have the relation $\mathbf{F} \ominus \mathbf{N}^6 = \bigcup_{i=1}^{3} (\mathbf{F} \ominus \mathbf{N}^4_i)$. Furthermore, the decomposition $\mathbf{F} = \bigcup_{i=1}^{3} \left\{ \bigcup_{k(i) \in K(i)} \mathbf{F}^2_{k(i)} \right\}$ derives the relation

$$
\begin{aligned}
\mathbf{F} \setminus (\mathbf{F} \ominus \mathbf{N}^6) &= \left(\bigcup_{i=1}^{3} \bigcup_{k(i) \in K(i)} \mathbf{F}^2_{k(i)} \right) \setminus \left\{ \bigcup_{i=1}^{3} ((\bigcup_{k(i) \in K(i)} \mathbf{F}^2_{k(i)}) \ominus \mathbf{N}^4_i) \right\} \\
(3.12) \qquad &= \bigcup_{i=1}^{3} \bigcup_{k(i) \in K(i)} \left\{ \mathbf{F}^2_{k(i)} \setminus (\mathbf{F}^2_{k(i)} \ominus \mathbf{N}^4_i) \right\}.
\end{aligned}
$$

$\square$

Theorem 5 implies the following properties.

Property 6. *On the surface of a grid Nef polyhedron, each point lies on two or three digital planes.*

Property 7. *The boundary $\partial\mathbf{F}$ of a three-dimensional digital object $\mathbf{F}$ is the union of the two-dimensional boundaries.*

Therefore, it is possible to construct $\partial\mathbf{F}$ from $\{\partial\mathbf{F}^2_{k(i)}\}^3_{k(i) \in K(i), i=1}$. Using this relation recursively, we can construct the boundary detection algorithm for three-dimensional digital objects from two-dimensional boundary detection algorithms.

4. Curvature Indices of Points

4.1. **Planar Curvature Indices.** Since we are concerned with a binary digital object, we affix values of 0 and 1 to points in the background and in objects, respectively. On $\mathbf{Z}^2$, the three types of point configurations illustrated in Figure 2.1 (d) exist in the neighbourhood of a point $\times$ on the boundary. In the three configurations A, B and C in Figure 2.1 (d), $\bullet$ and $\circ$ are points on the boundary and in the background, respectively. Setting $f_i \in \{0, 1\}$ to be the value of point $\boldsymbol{x}_i$ such that

$$(4.1) \qquad \begin{array}{ccc} & \boldsymbol{x}_3 = (m, n+1)^\top & \\ \boldsymbol{x}_5 = (m-1, n)^\top & \boldsymbol{x}_0 = (m, n)^\top & \boldsymbol{x}_1 = (m+1, n)^\top \\ & \boldsymbol{x}_7 = (m, n-1)^\top, & \end{array}$$

the curvature of point $\boldsymbol{x}_0$ is defined by

$$(4.2) \qquad r(\boldsymbol{x}_0) = 1 - \frac{1}{2} \sum_{k \in N} f_k + \frac{1}{4} \sum_{k \in N} f_k f_{k+1} f_{k+2},$$

where $N = \{1, 3, 5, 7\}$ and $k + 8 = k$ [24]. The curvature indices of configurations (a), (b) and (c) are positive, zero and negative, respectively. Therefore, we call these configurations convex, flat and concave, and affix the indices $+$, 0 and $-$, respectively.

4.2. **Spatial Curvature Indices.** Using combinations of planar curvature indices on three mutually orthogonal planes passing through a point $\boldsymbol{x}_0$, we define the curvature index of a point $\boldsymbol{x}_0$ in $\mathbf{Z}^3$ since the 6-neighbourhood is decomposed into three 4-neighbourhoods. On the boundary of a 6-connected object, Theorem 5 implies that there exist nine configurations in the $3 \times 3 \times 3$ local neighbourhoods shown in Figure 4.1.
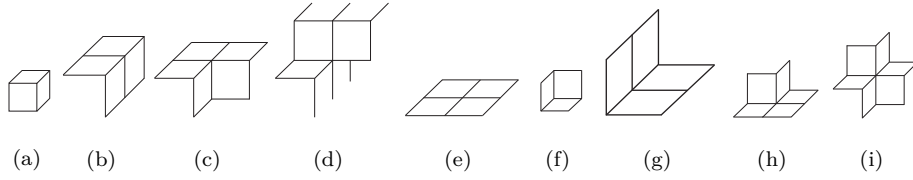

$$\text{(a)} \qquad \text{(b)} \qquad \text{(c)} \qquad \text{(d)} \qquad \text{(e)} \qquad \text{(f)} \qquad \text{(g)} \qquad \text{(h)} \qquad \text{(i)}$$

Figure 4.1: Local configurations of 2-manifold in $\mathbf{Z}^3$

Setting α_i to be the planar curvature index on plane $\mathbf{Z}_i^2(k(i))$ for $i = 1, 2, 3$, the curvature index of a point in $\mathbf{Z}^3$ is a triplet of two-dimensional curvature indices $(\alpha_1, \alpha_2, \alpha_3)$ such that $\alpha_i \in \{+, -, 0, \emptyset\}$. Here, if $\alpha_i = \emptyset$, the curvature index of a point on the plane $\mathbf{Z}_i^2(k(i))$ is not defined. Therefore, for the boundary points, seven configurations,

$$(4.3) \qquad \begin{matrix} (+, +, +), & (+, +, -), & (+, 0, 0), \\ & (0, 0, \emptyset), & \\ (-, -, -), & (+, -, -), & (-, 0, 0), \end{matrix}$$

and their permutations are possible [11]. For a spatial curvature index α, setting $n(\alpha)$ to be the number of octspaces in the $3 \times 3 \times 3$ neighbourhood of a point, we have the relations

$$(4.4) \qquad \begin{matrix} n((+, +, +)) = 1, & n((+, +, -)) = 3, & n((+, 0, 0)) = 2, \\ & n((0, 0, \emptyset)) = 4, & \\ n((-, -, -)) = 4, 7, & n((+, -, -)) = 4, 5, & n((-, 0, 0)) = 6. \end{matrix}$$

Since there exist two configurations for $(-, -, -)$ and $(+, -, -)$, we set

$$(4.5) \qquad n((-, -, -)) = 7, \ n((-, -, -)_-) = 4, \ n((+, -, -)) = 5, \ n((+, -, -)_+) = 4.$$

Furthermore, since $\frac{i}{8} = 1 - \frac{8-i}{8}$, the vertex angles of the nine configurations on the boundary are

$$(4.6) \qquad \begin{matrix} \gamma((+, +, +)) = \frac{1}{8}, & \gamma((+, 0, 0)) = \frac{2}{8}, & \gamma((+, +, -)) = \frac{3}{8}, \\[2mm] \gamma((+, -, -)_+) = \frac{4}{8}, & \gamma((0, 0, \emptyset)) = 0, & \gamma((-, -, -)_-) = \frac{-4}{8}, \\[2mm] \gamma((+, -, -)) = \frac{-3}{8}, & \gamma(-, 0, 0) = \frac{-2}{8}, & \gamma((-, -, -)) = \frac{-1}{8}. \end{matrix}$$

Definition 8. Using the nine configurations of eq. (4.6), we define the relation between the codes and the triplet vectors $\boldsymbol{\alpha}(\boldsymbol{x})$ for point $\boldsymbol{x}$ as

(4.7)

$\boldsymbol{\alpha}(\boldsymbol{x})$	$\gamma(\boldsymbol{x})$	$\boldsymbol{\alpha}(\boldsymbol{x})$	$\gamma(\boldsymbol{x})$
$(1,1,1)^\top$	$(0,0,+)$	$(1,1,-1)^\top$	$(+,+,-)$
	$(0,+,0)$		$(-,-,+)$
	$(+,0,0)$	$(1,-1,1)^\top$	$(+,-,+)$
	$(+,+,+)$		$(-,+,-)$
	$(-,-,-)$	$(-1,1,1)^\top$	$(-,+,+)$
	$(0,0,-)$		$(+,-,-)$
	$(0,-,0)$	$(1,1,0)^\top$	$(0,0,\emptyset)$
	$(-,0,0)$	$(1,0,1)^\top$	$(0,\emptyset,0)$
		$(0,1,1)^\top$	$(\emptyset,0,0)$.

Definition 9. Setting $s(\pm) = \pm 1$, $s(0) = 0$ and $s(\emptyset) = 0$, we define the vertex indices as

(4.8) $$f(\alpha_1, \alpha_2, \alpha_3) = (s(\alpha_1) + s(\alpha_2) + s(\alpha_3)) \times (|s(\alpha_1)| + |s(\alpha_2)| + |s(\alpha_3)|)$$

using eq. (4.7).

This function $f(\cdot, \cdot, \cdot)$ takes values of $\{0, \pm 1, \pm 3, \pm 9\}$ according to the configurations on the boundary. Furthermore, the sign of the function $f(\alpha_1, \alpha_2, \alpha_3)$ indicates the direction of normal vector $\boldsymbol{n}(\boldsymbol{x})$ at point $\boldsymbol{x}$ on the boundary. The direction of the normal vector is outward or inward if the sign is positive or negative, respectively.

Setting n_i to be the number of points whose vertex angles are $\frac{i}{8}$, we define three vectors,

(4.9) $$\boldsymbol{n} = (n_{-4}, n_{-3}, n_{-2}, n_{-1}, n_0, n_1, n_2, n_3, n_4)^\top$$

(4.10) $$\boldsymbol{a} = (-2, -1, 0, 1, 0, 1, 0, -1, -2)^\top,$$

(4.11) $$\boldsymbol{k} = \boldsymbol{a}^2 = (4, 1, 0, 1, 0, 1, 0, 1, 4)^\top.$$

Using these vectors, we have the following theorems for the Euler characteristics of digital objects [4, 2, 3, 5].

Theorem 10. *An object $\mathbf{F}$ without tunnels satisfies the relationship*

(4.12) $$\boldsymbol{a}^\top \boldsymbol{n} = 8.$$

Theorem 11. *A collection of objects $\{\mathbf{F}_i\}_{i=1}^n$, with g_i tunnels satisfies the relationship*

(4.13) $$\boldsymbol{a}^\top \boldsymbol{n} = 8 \sum_{i=1}^{n} (1 - g_i).$$

5. DIGITAL CURVATURE FLOW

5.1. **Normal Vectors on Digital Plane.** For $n = 2, 3$ we define the dual set for the set lattice points $\mathbf{Z}^n$ as

(5.1) $$\overline{\mathbf{Z}^n} = \{\boldsymbol{x} + \frac{1}{2}\boldsymbol{e} | \boldsymbol{x} \in \mathbf{Z}^n\},$$

where $\boldsymbol{e} = (1,1)^\top$ and $\boldsymbol{e} = (1,1,1)^\top$ for $n = 2$ and $n = 3$, respectively. We call $\mathbf{Z}^n$ and $\overline{\mathbf{Z}^n}$ the lattice and the dual lattice, respectively. Using the lattice and the dual lattice, we define the curvature flow for surfaces defined in the lattice space.

We first deal with a closed planar curve. Setting $\mathbf{C} = \{\boldsymbol{x}_j\}_{j=1}^N$ to be the set of points on the boundary $\partial \mathbf{F}_{k(i)}$ for fixed i and $K(i)$, we assume that only two points $\boldsymbol{x}_{j=1}$ and $\boldsymbol{x}_{j+1}$ are connected for each i and that the triplet $\boldsymbol{x}_{j-1}$, $\boldsymbol{x}_j$ and $\boldsymbol{x}_{j+1}$ lies in the anticlockwise direction on the boundary.

Using triplet of points $\boldsymbol{x}_{j-1}$, $\boldsymbol{x}_j$ and $\boldsymbol{x}_{j+1}$, we compute the normal vector of each point. For the planar point $\boldsymbol{x}_j = (x_j, y_j)^\top$, we express this vector as the complex number $z_j = x_j + iy_j$. Setting

(5.2) $$\alpha_j + i\beta_j = \left(\frac{z_{j+1} - z_j}{z_{j-1} - z_j} \right)^{\frac{1}{2}},$$

we define the outward normal vector $\boldsymbol{n}_j$ as

$$(5.3) \qquad \boldsymbol{n}_j = \lambda \boldsymbol{a}_j, \ \boldsymbol{a}_j = (\alpha_j, \beta_j)^\top,$$

for point $\boldsymbol{x}_j$. From the local configurations of triplets of vectors on the boundary, there exist three combinations for α_j and β_j:

(1) $|\alpha_j| = 1/\sqrt{2}$ and $|\beta_j| = 1/\sqrt{2}$,
(2) $|\alpha_j| = 0$ and $|\beta_j| \neq 0$,
(3) $|\alpha_j| \neq 0$ and $|\beta_j| = 0$.

These geometrical relations imply that

$$(5.4) \qquad \lambda_j = \begin{cases} \frac{1}{2} & \text{if } \alpha_j \beta_j = 0, \\ \frac{1}{\sqrt{2}} & \text{otherwise.} \end{cases}$$

Figure 5.1 shows the directions of normal vectors on a digital curve on a plane.

Figure 5.1: Normal vectors on the corner. On the digital boundary, there exist three configurations of the normal vectors of the point.

Using the curvature code $\gamma(\boldsymbol{x}_j)$ of each point $\boldsymbol{x}_j$, we classify points on the boundary $\mathbf{C} := \partial \mathbf{F}_{k(i)}$ for fixed i and $k(i)$ into types $\mathbf{N}_+$ and $\mathbf{N}_-$.

Definition 12. For a sequence of flat points

$$(5.5) \qquad \mathbf{N}_-(j) = \{\boldsymbol{x}_\beta | \gamma(\boldsymbol{x}_\beta) = 0, \ j < \beta < j + m, \ s(\gamma(\boldsymbol{x}_j)) \times s(\gamma(\boldsymbol{x}_{j+m})) = -1\},$$

where $s(\gamma(\boldsymbol{x})) = 1$ and $s(\gamma(\boldsymbol{x})) = -1$ for $(\gamma(\boldsymbol{x}_j) and \gamma(\boldsymbol{x}_{j+m})) = (+, -)$ and $(\gamma(\boldsymbol{x}_j) and \gamma(\boldsymbol{x}_{j+m})) = (-, +)$, respectively.

Each $\mathbf{N}_-(j)$ is a sequence of flat points whose one endpoint is the concave point. Then, we set

$$(5.6) \qquad \mathbf{N}_- = \bigcup_j \mathbf{N}_-(j), \ \mathbf{N}_+ = \mathbf{C} \setminus \mathbf{N}_-.$$

Since $\mathbf{N}_-$ is the union of sequences whose one endpoint is the concave point on the boundary, we have the relation

$$(5.7) \qquad \mathbf{C} = \mathbf{N}_+ \bigcup \mathbf{N}_-, \ \mathbf{N}_+ \bigcap \mathbf{N}_- = \emptyset.$$

Therefore, points in $\mathbf{N}_+$ and $\mathbf{N}_-$ lie in convex and concave parts on the original boundary, respectively.

Definition 13. If an end of a line segment is negative, we say that this line segment is a negative line segment. Furthermore, if both ends are positive, we say that this line segment is a positive line segment.

Using these geometrical properties of point sets $\mathbf{N}_+$ and $\mathbf{N}_-$ on the boundary, we define a global transform from point set $\mathbf{C}$ on $\mathbf{Z}^2$ to point set $\overline{\mathbf{C}}$ on $\overline{\mathbf{Z}^2}$ as

$$(5.8) \qquad \overline{\boldsymbol{x}}_j = \begin{cases} \boldsymbol{x}_j - \boldsymbol{n}_j, & \text{if } \gamma(\boldsymbol{x}_j) \neq 0, \\ \boldsymbol{x}_j - \boldsymbol{n}_j \pm \frac{1}{2}\boldsymbol{n}_j^\top, & \text{if } \gamma(\boldsymbol{x}_j) = 0, \ \boldsymbol{x}_j \in \mathbf{N}_+, \\ \boldsymbol{x}_j + \boldsymbol{n}_j \pm \frac{1}{2}\boldsymbol{n}_j^\top, & \text{if } \gamma(\boldsymbol{x}_j) = 0, \ \boldsymbol{x}_j \in \mathbf{N}_-. \end{cases}$$

5.2. **Normal Vectors in Digital Space.** Point configurations on the boundary are defined by the digital lines which lie on two or three mutually orthogonal vectors. Therefore, we define three normal vectors $\boldsymbol{n}_1$, $\boldsymbol{n}_2$ and $\boldsymbol{n}_3$ which lie on planes perpendicular to vectors $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and $\boldsymbol{e}_3$, respectively. According to these definitions, for $\boldsymbol{n} = (\alpha, \beta)^\top$ on each plane, we have

$$(5.9) \qquad \boldsymbol{n}_1 = (0, \alpha, \beta)^\top, \ \boldsymbol{n}_2 = (\alpha, 0, \beta)^\top, \ \boldsymbol{n}_3 = (\alpha, \beta, 0)^\top.$$

Using the vector $\boldsymbol{n}_\alpha$, $\alpha = 1, 2, 3$, we define the normal vector $\boldsymbol{n}(\boldsymbol{x})$ for a point $\boldsymbol{x}$ on the boundary $\mathbf{S}$ in digital space $\mathbf{Z}^3$. There are nine types of vertex configurations in a $3 \times 3 \times 3$ neighbourhood on the boundary. Using eq. (4.7), we define the normal vector $\boldsymbol{n}(\boldsymbol{x})$ of point $\boldsymbol{x}$.

Definition 14. The normal vector of a point $\boldsymbol{x} \in \mathbf{Z}^3$ is

$$(5.10) \qquad \boldsymbol{n}(\boldsymbol{x}) = \frac{1}{2} \boldsymbol{N} \boldsymbol{\alpha}(\boldsymbol{x})$$

for matrix $\boldsymbol{N} = (\boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}_3)$ on the surface of a Nef polyhedron.

5.3. **Curvature-based Motion of Points.** We derive a transformation using the global information of the concavity of the digital boundary.

Definition 15. For boundary points whose two-dimensional codes are zero, we affix the same two-dimensional curvature codes with these of line segments on which points lie.

The rule of Definition 15 derives the codes $(\alpha, \beta_1, \beta_2)$, $(\beta_1, \alpha, \beta_2)$ and $(\beta_1, \beta_2, \alpha)$, where $\beta_i \in \{+, -\}$, for a point whose codes are $(\alpha, 0, 0)$, $(0, \alpha, 0)$ and $(0, 0, \alpha)$, where $\alpha \in \{+, -, \emptyset\}$. These notations lead to the codes $(+, +, +)$, $(+, +, -)$, $(+, -, -)$, $(-, -, -)$, $(\emptyset, +, +)$, $(\emptyset, +, -)$, $(\emptyset, -, -)$ and their permutations. Using these codes, we define the code of the flat points and points on edges as

$$(5.11) \qquad g(\alpha, \beta_1, \beta_2) = \begin{cases} +, & \text{if } f(\alpha, \beta_1, \beta_2) = 9, \\ -, & \text{otherwise}, \end{cases}$$

for $(\alpha, \beta_1, \beta_2)$ and its permutations.

Setting

$$(5.12) \qquad h(\beta_1, \beta_2) = (s(\beta_1) + s(\beta_2)) \times (|s(\beta_1)| + |s(\beta_2)|),$$

we set the codes of flat points and points on edges as positive and negative if $h(\beta_1, \beta_2) > 0$ and otherwise, respectively. For these curvature codes, the code of a flat point is defined as negative if the number of negative codes in the curvature code is positive. Furthermore, we move flat points outward and inward, if the codes are positive and negative, respectively.

Definition 16. For two line segments, which are mutually orthogonal, pass through flat points with codes $(\emptyset, 0, 0)$, $(0, \emptyset, 0)$ and $(0, 0, \emptyset)$, using the codes of the end points of these two line segments on a plane, we can affix the codes of flat points. Furthermore, If at least one end point is negative, we assign the negative code to a point on this line.

There are six possibilities for the configurations of end points of two mutually orthogonal line segments. These definitions for codes also conclude that the codes of points with the curvature codes $(+, 0, 0)$, $(0, +, 0)$ and $(0, 0, +)$ on an edge are negative if one end or both ends are negative. Figure 5.2 shows examples of the signs of vertex indices on the boundary.

The normal vector on the discrete surface defined by eq. (5.10) is in the form

$$(5.13) \qquad \boldsymbol{n}(\boldsymbol{x}) = \frac{1}{2}(a_1 \boldsymbol{e}_1 + a_2 \boldsymbol{e}_2 + a_3 \boldsymbol{e}_3),$$

where $a_i \in \{-1, 0, 1\}$. Therefore, if $a_i = 1$ and $a_i = -1$, the vector $(\boldsymbol{x} + \boldsymbol{n}(\boldsymbol{x}))$ determines the transformation from $\boldsymbol{x} \in \mathbf{Z}^3$ to $\boldsymbol{y} \in \overline{\mathbf{Z}}^3$. Moreover, if and only if the codes are $(+, +, +)$, $(-, -, -)$, and $(-, -, -)_-$ and (α, β, γ) for $\alpha, \beta, \gamma \in \{+, -\}$, the vector $(\boldsymbol{x} + \boldsymbol{n}(\boldsymbol{x}))$ determines the transformation from $\boldsymbol{x} \in \mathbf{Z}^3$ to $\boldsymbol{y} \in \overline{\mathbf{Z}}^3$.
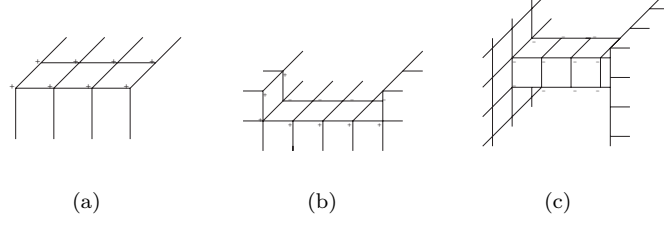
Figure 5.2: Signs for flat points and edges.

For the codes $(-,-,-)$, $(\alpha,0,0)$, $(\alpha,0,0)$ and $(0,0,\alpha)$ for $\alpha \in \{+,-,\emptyset\}$, the normal vectors hold the relations

(5.14)

$\alpha(\boldsymbol{x})$	$(\emptyset,0,0)$	$(0,\emptyset,0)$	$(0,0,\emptyset)$	$(-,-,-)$
$\boldsymbol{n}(\boldsymbol{x})$	$\boldsymbol{0}$	$\boldsymbol{0}$	$\boldsymbol{0}$	$a_i\boldsymbol{e}_i + a_j\boldsymbol{e}_j$

$\alpha(\boldsymbol{x})$	$(\pm,0,0)$	$(0,\pm,0)$	$(0,0,\pm)$
$\boldsymbol{n}(\boldsymbol{x})$	$\pm\frac{1}{2}(a_2\boldsymbol{e}_2 + a_3\boldsymbol{e}_3)$	$\pm\frac{1}{2}(a_1\boldsymbol{e}_1 + a_3\boldsymbol{e}_3)$	$\pm\frac{1}{2}(a_1\boldsymbol{e}_1 + a_2\boldsymbol{e}_2)$.

However, these normal vectors do not define the transformation from points in $\mathbf{Z}^3$ to $\overline{\mathbf{Z}^3}$. Therefore, we define the transformation from $\boldsymbol{x} \in \mathbf{Z}^3$ to $\boldsymbol{y} \in \overline{\mathbf{Z}^3}$ using $g(\cdot,\cdot,\cdot)$.

Definition 17. The point transformation $S(\cdot)$ between $\boldsymbol{x} \in \mathbf{Z}^3$ and $\boldsymbol{y} \in \mathbf{Z}^3$ is defined as

(5.15)
$$\boldsymbol{y} = S(\boldsymbol{x}) = \boldsymbol{x} + \epsilon\boldsymbol{m}(\boldsymbol{x})$$

for

(5.16)

$\alpha(\boldsymbol{x})$	$(\emptyset,0,0)$	$(0,\emptyset,0)$	$(0,0,\emptyset)$	$(-,-,-)_+$
$\boldsymbol{m}(\boldsymbol{x})$	$\pm\frac{1}{2}\boldsymbol{e}_1$	$\boldsymbol{n}(\boldsymbol{x}) \pm \frac{1}{2}\boldsymbol{e}_2$	$\boldsymbol{n}(\boldsymbol{x}) \pm \frac{1}{2}\boldsymbol{e}_3$	$\frac{1}{2}\boldsymbol{n}(\boldsymbol{x}) \pm \frac{1}{2}\boldsymbol{e}_k$

$\alpha(\boldsymbol{x})$	$(\pm,0,0)$	$(0,\pm,0)$	$(0,0,\pm)$
$\boldsymbol{m}(\boldsymbol{x})$	$\pm\frac{1}{2}(a_2\boldsymbol{e}_2 + a_3\boldsymbol{e}_3)$	$\pm\frac{1}{2}(a_1\boldsymbol{e}_1 + a_3\boldsymbol{e}_3)$	$\pm\frac{1}{2}(a_1\boldsymbol{e}_1 + a_2\boldsymbol{e}_2)$,

where

(5.17) $\quad \epsilon = \begin{cases} 1, & \text{for } (+,+,+),\ (-,-,-),\ (-,-,-)_-,\ (+,+,-),\ (+,-,+) \text{ and } (-,+,+), \\ -1, & \text{for points where } g(\alpha,\beta_1,\beta_2) < 0. \end{cases}$

The negative sign of ϵ depends on the configuration of two mutually orthogonal line segments which pass through the points. Since the saddle points on a discrete dumbbell move inward, our method preserves the topology of the dumbbell. However, according to the definitions of the directions of the motion of points in the curvature flow, positive points on edges with both end points negative move outward. This configuration defines the outward motion for points on the bar of a dumbbell. Therefore, these definitions of the codes of flat points and edges preserve the topology of dumbbells. The successive application of eq. (5.15) defines a transformation of point sets between the lattice and the dual lattice.

Definition 18. The digital curvature flow is a sequence of point sets $\mathbf{S}_m$ for $m \geq 0$ such that

(5.18)
$$\mathbf{S}_m = \begin{cases} \partial\mathbf{F} \subset \mathbf{Z}^3 & \text{if } m = 0, \\ \{\boldsymbol{x} | \boldsymbol{x} = S(\overline{\boldsymbol{x}}), \overline{\boldsymbol{x}} \in \overline{\mathbf{Z}^3}\} \subset \mathbf{Z}^3 & \text{if } m = 2k \text{ for } k \geq 1, \\ \{\overline{\boldsymbol{x}} | \overline{\boldsymbol{x}} = S(\boldsymbol{x}), \boldsymbol{x} \in \mathbf{Z}^3\} \subset \overline{\mathbf{Z}^3} & \text{otherwise.} \end{cases}$$

The odd and even steps of the digital curvature flow transform points on the lattice to points on the dual lattice and points on the dual lattice to points on the lattice, respectively.

Now, we show an example.

Example 19. Setting lattice points on a polyhedron to be

$$
\mathbf{P}_{10} = \left\{ \begin{array}{lll} (0,0,2)^\top, & (0,1,2)^\top, & \\ (0,0,1)^\top, & (0,1,1)^\top, & (0,2,1)^\top \\ (0,0,0)^\top, & (0,1,0)^\top, & (0,2,0)^\top \end{array} \right\},
$$

(5.19)
$$
\mathbf{P}_{11} = \left\{ \begin{array}{lll} (1,0,2)^\top, & (1,1,2)^\top, & \\ (1,0,1)^\top, & (1,1,1)^\top, & (1,2,1)^\top \\ (1,0,0)^\top, & (1,1,0)^\top, & (1,2,0)^\top \end{array} \right\}
$$

$$
\mathbf{P}_{12} = \left\{ \begin{array}{ll} (2,0,1)^\top, & (2,1,1)^\top \\ (2,0,0)^\top, & (2,1,0)^\top \end{array} \right\},
$$

the digital curvature flow

(5.20)
$$
\begin{array}{rcl}
(\frac{1}{2},\frac{1}{2},\frac{1}{2})^\top & \leftarrow & (0,0,0)^\top, \\
(\frac{1}{2},\frac{3}{2},\frac{1}{2})^\top, & \leftarrow & (0,2,1)^\top, \quad (0,2,0)^\top, \\
& & (1,2,1)^\top, \quad (1,2,0)^\top \\
(\frac{3}{2},\frac{3}{2},\frac{1}{2})^\top & \leftarrow & (1,1,0)^\top \\
(\frac{3}{2},\frac{1}{2},\frac{1}{2})^\top, & \leftarrow & (1,0,0)^\top, \quad (2,0,0)^\top, \quad (2,1,0)^\top, \\
& & (2,1,1)^\top, \quad (2,0,1)^\top \\
(\frac{1}{2},\frac{1}{2},\frac{3}{2})^\top, & \leftarrow & (0,0,2)^\top, \quad (0,1,2)^\top, \\
& & (1,0,2)^\top, \quad (1,1,2)^\top, \\
(\frac{1}{2},\frac{3}{2},\frac{3}{2})^\top & \leftarrow & (0,1,1)^\top, \\
(\frac{3}{2},\frac{3}{2},\frac{3}{2})^\top & \leftarrow & (1,1,1)^\top, \\
(\frac{3}{2},\frac{1}{2},\frac{3}{2})^\top & \leftarrow & (1,0,1)^\top
\end{array}
$$

derives the cube

(5.21)
$$
\mathbf{P} = \left\{ \begin{array}{lll} (\frac{1}{2},\frac{1}{2},\frac{1}{2})^\top & (\frac{1}{2},\frac{3}{2},\frac{1}{2})^\top & (\frac{3}{2},\frac{3}{2},\frac{1}{2})^\top \\ (\frac{3}{2},\frac{1}{2},\frac{1}{2})^\top & (\frac{1}{2},\frac{1}{2},\frac{3}{2})^\top & (\frac{1}{2},\frac{3}{2},\frac{3}{2})^\top \\ (\frac{3}{2},\frac{3}{2},\frac{3}{2})^\top & (\frac{3}{2},\frac{1}{2},\frac{3}{2})^\top & \end{array} \right\}.
$$

In the next step, $\mathbf{P}$ converges to the point $(1,1,1)^\top$. Figures 5.3 (a) and (b)respectively show the original polyhedron and a cube obtained as the result of the digital curvature flow.
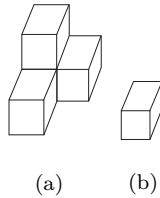

(a)   (b)

Figure 5.3: Example of deformation. (a) is transformed to (b). Furthermore, In the next step, $\mathbf{P}$ converges the point $(1,1,1)^\top$

The curvature flow transforms each planar curve segment passing through lattice points on a polyhedral boundary to a straight line segment passing through lattice points on a polyhedral boundary. Furthermore, each closed curve on a plane is transformed to a rectangle. If $\gamma(\boldsymbol{x}) = (+,+,+)$ at each step, the digital curvature flow eliminates 3/4 unit area from the corners. Moreover, positive and negative parts on the boundary move outward and inward, respectively. These considerations conclude the following theorems.

Theorem 20. *For a closed surface, in each step, the digital curvature flow shrinks the boundary on each planar 6 unit area from corners such that $\gamma(\boldsymbol{x}) = (+,+,+)$.*

Theorem 21. *The digital curvature flow preserves the topology if there is no tunnel.*

Theorem 22. *The digital curvature flow transforms a boundary to a cuboid.*

Theorem 23. *The final form of the digital curvature flow is a spatial rectangle.*

Furthermore, setting the surface energy.

$$(5.22) \qquad\qquad E = \boldsymbol{k}^\top \boldsymbol{n},$$

we have the following theorem.

Theorem 24. *On a closed two-dimensional manifold $\partial\mathbf{F}$ in $\mathbf{Z}^3$*

$$(5.23) \qquad\qquad E = \boldsymbol{k}^\top \boldsymbol{n} \geq 8(1+g).$$

(*Proof*) From theorems 20, 21, 22 and 23, if $\mathbf{F}$ has no hole, $\mathbf{F}$ is topologically equivalent to a cuboid. Since $E = 8$ for a cuboid, $E \geq 8$ for an object without holes. Furthermore, if an object has g holes, the object is topologically equivalent to a cuboid with g cuboid tunnels. Therefore, for an object with g tunnels, $E \geq 8(1 + g)$. $\qquad\square$

Moreover, we have the following theorem for E from theorem 20.

Theorem 25. *For the surface evolution of eq. (5.15), $E = \boldsymbol{k}^\top \boldsymbol{n}$ satisfies the relation*

$$(5.24) \qquad\qquad \frac{\partial E}{\partial t} \leq 0.$$

6. Conclusions

We have defined curvature codes on a digital manifold in the three-dimensional digital space $\mathbf{Z}^3$. A point on a 6-connected three-dimensional digital manifold lies on at least two 4-connected planar curves, which lie on a pair of perpendicular digital planes. From this geometrical configuration of orthogonal slices of digital manifolds, the three-dimensional curvature codes were constructed as a combination of three planar curvature codes.

As an extension, we can define the curvature codes of points on an m-dimensional digital manifold in the n-dimensional digital space for $m \leq n - 1$. For example, if $m = n - 1$, the curvature code on the digital manifold is described in the form $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_n)$, where $\alpha_i \in \{+1, -1, 0, \emptyset\}$ and the curvature index of this point is

$$(6.1) \qquad f(\alpha_{\pi(1)}, \alpha_{\pi(2)}, \cdots \alpha_{\pi(n)}) = (\sum_{i=1}^{n} s(\alpha_{\pi(i)})) \times (\Pi_{i=1}^{n}|s(\alpha_{\pi(i)})|).$$

Theorem 11, which characterizes the topology of closed surfaces in the three-dimensional digital surface, is a digital version of the Gauss-Bonnet theorem. This theorem suggests that using the curvature codes on a higher-dimensional digital manifold, the digital version of the Chern-Gauss-Bonnet theorem can be constructed as

$$(6.2) \qquad\qquad \boldsymbol{a}^\top \boldsymbol{n} = 2^n(1 - g),$$

where $\boldsymbol{a}$ and $\boldsymbol{n}$ are vectors which define the weights for the configurations of points on the surface and the number of configurations on the surface, respectively, and g is the number of the holes of an object. Moreover, the surface energy

$$(6.3) \qquad\qquad E = \boldsymbol{k}^\top \boldsymbol{n} = 2^n(1 + g),$$

where each element of the vector $\boldsymbol{k}$ is the square of the corresponding element of $\boldsymbol{a}$, will satisfy the inequality $\frac{\partial E}{\partial t} \leq 0$.

References

[1] Marshall Bern and David Eppstein. Mesh generation and optimal triangulation. *Computing in Euclidean Geometry, 2nd Edition*, pages 47–123, 1995.

[2] Hanspeter Bieri and Walter Nef. A recursive sweep-plane algorithm, determining all cells of a finite division of R^d. *Computing*, 28:189–198, 1982.

[3] Hanspeter Bieri and Walter Nef. A sweep-plane algorithm for computing the volume of polyhedra represented in Boolean form. *Linear Algebra and Its Applications*, 52/53:69–97, 1983.

[4] Hanspeter Bieri and Walter Nef. Algorithms for the euler characteristic and related additive functionals of digital objects. *CVGIP*, 28:166–175, 1984.

[5] Hanspeter Bieri and Walter Nef. A sweep-plane algorithm for computing the euler-characteristic of polyhedra represented in boolean form. *Computing*, 34:287–304, 1985.

[6] Alexander I. Bobenko and Yuri B. Suris. *Discrete Differential Geometry: Integrable Structure*. American Mathematical Society, 2008.

[7] Alfred M. Bruckstein, Guillermo Shapiro, and Doron Shaked. Evolution of planar polygons. *J. Pattern Recognition and Artificial Intelligence*, 9:991–1014, 1995.

[8] Bastien Chopard and Michel Droz. *Cellular Automata Modeling of Physical Systems*. Cambridge University Press, Cambridge, 1998.

[9] Nira Dyn, David Levinand, and Samuel Rippa. Data dependent triangulations for piecewise linear interpolation. *IMA J. Numerical Analysis*, 10:137–154, 1990.

[10] Gerhard Huisken. Flow by mean curvature of convex surface into sphere. *J. Differential Geometry*, 20:237–266, 1984.

[11] Atsushi Imiya. Geometry of three-dimensional neighbourhood and its applications (in Japanese). *Trans. of Information Processing Society of Japan*, 34:2153–2164, 1993.

[12] Yukiko Kenmochi and Atsushi Imiya. Deformation of discrete object surfaces. *Lecture Notes in Computer Science*, 1296:146–153, 1997.

[13] Ron Kimmel. *Numerical Geometry of Images: Theory, Algorithms, and Applications*. Springer, Heidelberg, 2007.

[14] Reinhard Klette and Azriel Rosenfeld. *Digital Geometry: Geometric Methods for Digital Picture Analysis*. Morgan Kaufmann, 2004.

[15] C.-N Lee, T. Poston, and Azriel Rosenfeld. Holes and genus of 2D and 3D digital images. *CVGIP*, 55:20–47, 1993.

[16] Douglas Lind and Brian Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, Cambridge, 1995.

[17] Tony Lindeberg. *Scale-Space Theory*. Kluwer Academic Publishers, Dordrecht, 1994.

[18] Tony Lindeberg. Generalized axiomatic scale-space theory. *Advances in Imaging and Electron Physics*, 178:1–96, 2013.

[19] Atsuyuki Okabe, Barry Boots, and Kokichi Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. John Wiley& Sons, Chichester, 1992.

[20] Samuel Rippa. Minimal roughness property of the Delaunay triangulation. *Computer Aided Geometric Design*, 7:489–497, 1990.

[21] Tomoya Sakai, Masaki Narita, Takuto Komazaki, Haruhiko Nishiguchi, and Atsushi Imiya. Image hierarchy in gaussian scale space. *Advances in Imaging and Electron Physics*, 165:175–263, 2011.

[22] Guillermo Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, Cambridge, 2001.

[23] James A. Sethian. *Level Set Methods: Evolving Interfaces in Geometry Fluid Mechanics, Computer Vision, and Material Science*. Cambridge University Press, Cambridge, 1996.

[24] Junichiro Toriwaki. *Digital Image Processing for Image Understanding, Vols.1 and 2 (in Japanese)*. Syokodo, Tokyo, 1988.

[25] Junichiro Toriwaki, Sigeki Yokoi, T. Yonekura, and T. Fukumura. Topological properties and topological preserving transformation of a three-dimensional binary picture. In *Proc. of the 6th ICPR*, pages 414–419, 1982.

[26] Junichiro Toriwaki and Hiroyuki Yoshida. *Fundamentals of Three-dimensional Digital Image Processing*. Springer, Heidelberg, 2009.

[27] Andrea Toselli and Olof Widlund. *Domain Decomposition Methods - Algorithms and Theory*. Springer, Heidelberg, 2005.

[28] Richard S. Varge. *Matrix Iterative Analysis, 2nd rev. and exp. ed.* Springer, Heidelberg, 2000.

[29] Joachim Weickert. *Anisotropic Diffusion in Image Processing, ECMI Series*. Teubner-Verlag, Stuttgart, 1998.

[30] Stephan Wolfram. *A New Kind of Science*. Wolfram Media, Champaign, 2002.

[31] T. Yonekura, Junichiro Toriwaki, T. Fukumura, and Sigeki Yokoi. On connectivity and the euler number of three-dimensional digitized binary picture. *Trans. of IECE Japan*, E63:815–816, 1980.

Supercomputing Laboratory, Institute of Management and Information Technologies, Chiba University, Yayoi-cho 1-33, Inage-ku, Chiba, 263-8522, Chiba, Japan • imiya@facaulty.chiba-u.jp

Digital shapes, digital boundaries and rigid transformations: A topological discussion

Yukiko Kenmochi, Phuc Ngo, Nicolas Passat, and Hugues Talbot

Abstract

Curvature is a continuous and infinitesimal notion. These properties induce geometrical difficulties in digital frameworks, and the following question is naturally asked: "How to define and compute curvatures of digital shapes?" In fact, not only geometrical but also topological difficulties are also induced in digital frameworks. The – deeper – question thus arises: "Can we still define and compute curvatures?" This latter question, that is relevant in the context of digitization, *i.e.*, when passing from $\mathbb{R}^n$ to $\mathbb{Z}^n$, can also be stated in $\mathbb{Z}^n$ itself, when applying geometric transformations on digital shapes. This paper proposes a preliminary discussion on this topic.

1. Introduction

In the continuous domain, the computation of curvature requires that the considered shapes – and more precisely their boundaries – present certain good properties, mostly in terms of differentiability. When passing from the continuous universe ($\mathbb{R}^n$) to the discrete one ($\mathbb{Z}^n$), the handling of curvature becomes much more complex. It is easy to guess that the induced difficulties derive from the necessity to model infinitesimal properties – namely the differentiability of boundaries – into a finite framework.

However, even before considering such geometrical concerns, it is crucial to keep in mind that there also exist topological concerns. Indeed, beyond its putative differentiability, the notion of boundary itself often becomes ill-defined in $\mathbb{Z}^n$. In other words, even if we consider a continuous shape in $\mathbb{R}^n$, whose boundary is an object of dimension $n-1$ and also an $(n-1)$-manifold, it is unfortunately infrequent that the boundary of its digitized analogue in $\mathbb{Z}^n$ is a discrete hypersurface, and a fortiori a discrete $(n-1)$-manifold.

During the last decades, some efforts were devoted to tackle this issue in the context of digitization. More precisely, some conditions were provided to guarantee the preservation of good geometrical and topological properties of shape boundaries, when passing from $\mathbb{R}^n$ to $\mathbb{Z}^n$. However, if we now know how to correctly handle curvature during this digitization step, it remains challenging to also define adequate conditions for curvature definition and analysis when processing digital shapes. In particular, it is difficult to preserve correct topological – and thus geometrical – properties of digital shape boundaries when applying geometric transformations, even the most simple such as rigid transformations.

In this paper – that is mainly related to the works published in [7, 8] – we expose some preliminary results devoted to this question. More precisely, we focus on the specific case of digital shapes defined on $\mathbb{Z}^2$, and on their behaviour under rigid transformations. Considering such a low dimension and such simple transformations may seem meaningless and irrelevant at a time when the hot topics are related to high-dimension objects under arbitrary deformations. Nevertheless,
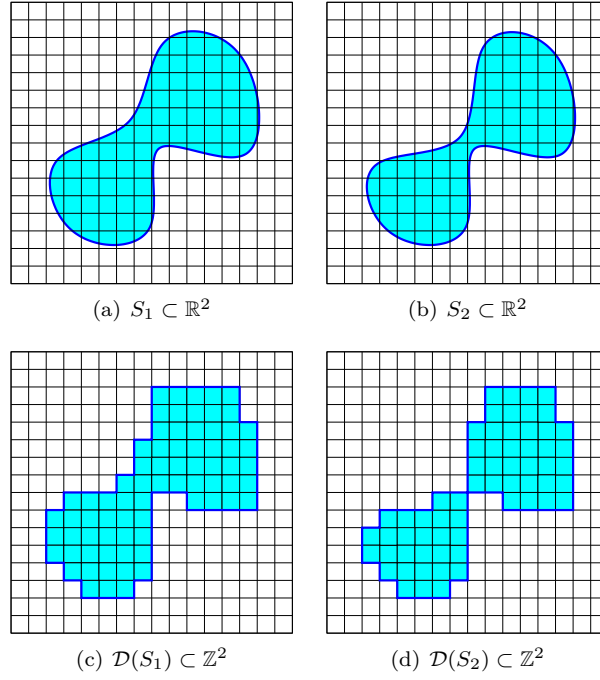
Figure 2.1: (a,b) Continuous shapes S_1 and S_2 in $\mathbb{R}^2$ (in cyan) and their boundaries (in blue). (c,d) The associated digital shapes $\mathcal{D}(S_1)$ and $\mathcal{D}(S_2)$ in $\mathbb{Z}^2$ (in cyan), and their digital boundaries (in blue). (c) The digital boundary of $\mathcal{D}(S_1)$ is a 1-manifold. (d) The digital boundary of $\mathcal{D}(S_2)$ is not a 1-manifold, by contrast with that of S_2.

beyond this apparent triviality, we show that the induced issues are not straightforward, and we intend to propose sound foundations for further developments at higher dimensions and for more general transformations.

2. Digital shapes and their boundaries

Let us consider a finite closed set S in $\mathbb{R}^2$ whose boundary is a (set of) 1-manifold(s) as an original shape. Since computers handle only finite structures, such a continuous shape S is represented as a digital image, *i.e.*, a finite set of pixels associated to points of $\mathbb{Z}^2$. The induced digital shape is denoted by $\mathcal{D}(S)$, referring to the digitization procedure $\mathcal{D}$ that allows us to pass from $\mathbb{R}^2$ to $\mathbb{Z}^2$.

There exist several models for $\mathcal{D}$ [2]. For instance, if we consider the Gaussian model, we obtain $\mathcal{D}(S) = S \cap \mathbb{Z}^2$; in other words, the digital shape $\mathcal{D}(S)$ of S is simply obtained by "sampling" S with respect to the regular structure of $\mathbb{Z}^2$. We will note $\overline{\mathcal{D}(S)}$ the complement of $\mathcal{D}(S)$ in $\mathbb{Z}^2$.

It is mandatory to provide an explicit and sound definition for the notion of boundary of a digital shape $\mathcal{D}(S)$. To this end, let us first consider the links that exist between the points of $\mathbb{Z}^2$ and the pixels of a digital image. A pixel P, associated to a point $\mathbf{x}$ of $\mathbb{Z}^2$, can be seen as a unit square of $\mathbb{R}^2$ centered on $\mathbf{x}$. In other words, we have $P = \mathbf{x} + [-1/2, 1/2]^2 \subset \mathbb{R}^2$. From a structural point of view, the pixels of a digital image are nothing but the Voronoi cells of $\mathbb{R}^2$ induced by the points of $\mathbb{Z}^2$. In particular, some couples of pixels share a part of their (continuous) boundaries. More precisely, two pixels P_1 and P_2 associated to $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively, satisfy this assertion iff their exists an edge between $\mathbf{x}_1$ and $\mathbf{x}_2$ in the (dual) Delaunay diagram associated to the above Voronoi diagram.

Based on these considerations, the boundary $\partial \mathcal{D}(S)$ of the digital shape $\mathcal{D}(S)$, also called the digital boundary for short, is straightforwardly associated to the continuous boundary induced by the pixels of $\mathcal{D}(S)$. Indeed, $\partial \mathcal{D}(S)$ can be modeled by a set of couples of points $(\mathbf{x}, \overline{\mathbf{x}}) \in \mathcal{D}(S) \times \overline{\mathcal{D}(S)}$, that share an edge in the Delaunay diagram of $\mathbb{Z}^2$.
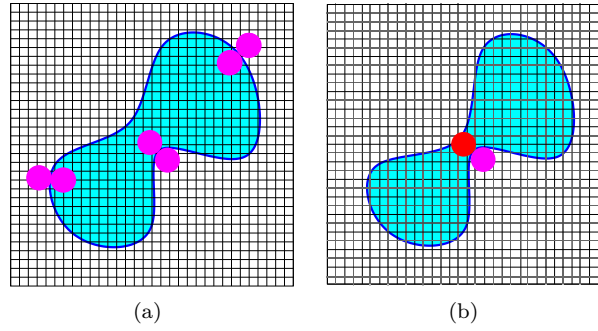
Figure 3.1: (a) An *r*-regular shape (in cyan). Some inner and outer *r*-radius open balls are depicted in magenta, with *r* higher than the pixel size of the "topology-preserving" digitization grid. (b) A shape that is not *r*-regular (in cyan). An example of *r*-radius open balls couple is depicted, where the "inner" ball (in red) does not entirely lie into the shape.

The handling of digital boundaries can be considered from a topological point of view. To this end, we can use the standard notion of neighbourhood stated in digital topology [3]. The *k*-neighbourhood of a point $\mathbf{x} \in \mathbb{Z}^2$ is defined by $N_k(\mathbf{x}) = \{\mathbf{y} \in \mathbb{Z}^2 \mid \|\mathbf{x} - \mathbf{y}\|_p = 1\}$, for $k = 4, 8$ where $p = 1, \infty$, respectively. Then, the boundary of $\mathcal{D}(S)$ is defined by $\partial\mathcal{D}(S) = \{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \in \mathcal{D}(S), \mathbf{y} \in \overline{\mathcal{D}(S)}, \mathbf{y} \in N_4(\mathbf{x})\}$.

Figure 2.1 illustrates two examples of digitization procedure, where it is easily seen that the topology of a digital shape is not always the same as that of the initial continuous shape. More precisely, we observe that digital shape boundaries are not always guaranteed to be 1-manifolds, even though the original shape boundaries are.

3. Digitization and topology preservation

The issue of topological alteration of shape boundaries during the digitization process has been considered in the literature. In particular, Latecki *et al.* defined some conditions for guaranteeing boundary integrity, based on two key notions, namely *r*-regularity and well-composedness.

Definition 1 (*r*-regularity [1])**.** A closed set $S \subset \mathbb{R}^2$ is *r-regular* if for each boundary point of S, there exist two tangent open balls of radius r, lying entirely in S and its complement $\overline{S}$, respectively.

This definition derives from classical concepts of differential geometry, namely osculating balls and normal vectors. By considering the class of *r*-regular shapes in $\mathbb{R}^2$, the condition for preserving topological properties – especially in terms of boundaries – between a continuous shape and its digital counterpart is the following.

Proposition 2 ([4])**.** *An r-regular set $S \subset \mathbb{R}^2$ has the same topological structure as its digitized version $\mathcal{D}(S)$, for pixels of size $d < r$.*

This result is indeed an extension of the compatibility property between S and $\mathcal{D}(S)$, presented by Pavlidis in [9]. In particular, Latecki *et al.* were driven by more pragmatic motivations, related to some sampling devices for image acquisition, like CCD cameras. Figure 3.1 provides an example and a counterexample of *r*-regular shapes.

In [4], it was also shown that the (topology-preserving) digitization process of an *r*-regular shape must yield a well-composed shape [5], whose definition relies on the following concepts of adjacency, connectedness and connected components, in digital topology [3]. Let X be a digital shape in $\mathbb{Z}^2$. We say that two distinct points $\mathbf{x}, \mathbf{y}$ of X are *k*-adjacent (for $k = 4, 8$) if $\mathbf{x} \in N_k(\mathbf{y})$ (and – equivalently – $\mathbf{y} \in N_k(\mathbf{x})$). From the induced (symmetric) *k*-adjacency relation on X, we obtain, by reflexive-transitive closure, the (equivalence) *k*-connectedness relation. The *k*-connected components of X are the equivalence classes for this relation. From these notions, we can define the notion of well-composedness.
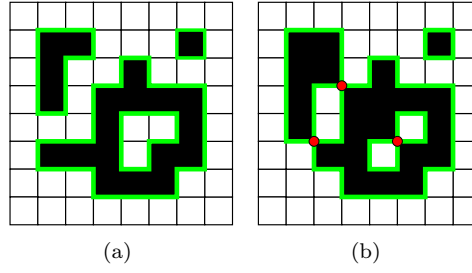
(a) (b)

Figure 3.2: (a) A well-composed shape X_1 of $\mathbb{Z}^2$ (in black). Its boundary ∂X_1 (in green) is a 1-manifold. (b) A shape X_2 of $\mathbb{Z}^2$ (in black) that is not well-composed. Its boundary ∂X_2 (in green) is not a 1-manifold (see the red dots).

Definition 3 (Well-composedness [5])**.** A digital shape X in $\mathbb{Z}^2$ is well-composed if each 8-connected component of X and of its complement $\overline{X}$ is also a 4-connected component.

Based on this definition, it is plain that the boundary ∂X of a digital shape X is a 1-manifold whenever X is well-composed. Some examples and counter-examples of well-composed shapes are provided in Figure 3.2.

As stated above, there actually exists a strong link between r-regularity and well-composedness.

Theorem 4 ([4])**.** *If S is r-regular, then $\mathcal{D}(S)$ is well-composed.*

Consequently, a continuous shape S of $\mathbb{R}^2$ that is r-regular – and whose boundary is a continuous 1-manifold that authorizes curvature analysis – still presents good properties after digitization, since its digital counterpart $\mathcal{D}(S)$ is well-composed and then also presents as border $\partial \mathcal{D}(S)$ a 1-manifold.

4. Rigid transformations of digital shapes

As stated above, in the digital framework, curvature analysis makes sense when considering shapes that are well-composed. In the continuity of this result, the question that we consider now is the following: "What are the conditions for allowing curvature analysis not only on a digital shape X but also on its image by a geometric transformation?". In particular, we focus on the most simple – yet non-trivial – case of rigid transformations.

In $\mathbb{R}^2$, a rigid transformation is a function

(4.1)
$$\left| \begin{array}{rcl} \mathcal{T} & : & \mathbb{R}^2 \rightarrow \mathbb{R}^2 \\ & & \mathbf{x} \mapsto \mathbf{R}.\mathbf{x} + \mathbf{t} \end{array} \right.$$

where $\mathbf{R}$ is a rotation matrix, *i.e.*, an element of the group $SO(2)$, and $\mathbf{t} \in \mathbb{R}^2$ is a translation vector. The rigid transformation $\mathcal{T}$ is a bijection, and its inverse function $\mathcal{T}^{-1}$ is also a rigid transformation.

Based on these definitions, the digital rigid transformations consist of composing the continuous rigid transformations with the standard rounding operator. We note $\mathcal{RIG}_{\mathbb{Z}^2}$ the set of all the digital rigid transformations.

Two transformation models can be considered for a digital rigid transformation T associated to a transformation $\mathcal{T}$. The Lagrangian (forwards) model consists of computing the direct image of the digital shape X by the transformation. However, as T is often neither injective nor surjective, this leads to topological defects. The Eulerian (backwards) model consists of computing the transformed image X_T as the shape whose image by the digital analogue T^{-1} of the inverse function $\mathcal{T}^{-1}$ of $\mathcal{T}$, lies into X. This is more satisfactory, since $\mathcal{T}^{-1}$ is defined on the whole transformed space $\mathbb{Z}^2$. In this model, that we consider hereafter, we have

(4.2)
$$X_T = D(\mathcal{T}(X \oplus \square))$$

where $\oplus$ is the classical dilation operator defined in mathematical morphology [6, Ch. 1], and $\square$ is the unit square of $\mathbb{R}^2$, centered on the origin. These relationships with well-known concepts of mathematical morphology are actually not a coincidence, as it will be discussed later.
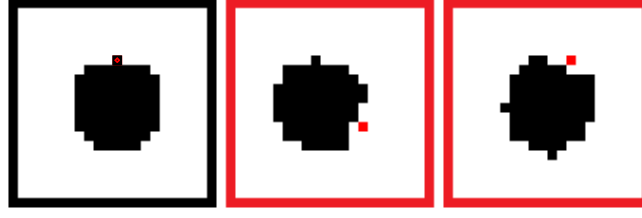
Figure 4.1: A digital disk (left) and its rigid transformations (center and right). The original digital shape is well-composed while the transformed ones are not well-composed.
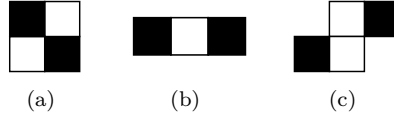

Figure 5.1: Forbidden patterns in regular shapes (up to $\pi/2$ rotations, symmetries and value inversion).

Unfortunately, the Eulerian model is not exempt from topological difficulties. In particular, the family of well-composed shapes is not stable with respect to $\mathcal{RIG}_{\mathbb{Z}^2}$. In other words, the transformed shape X_T obtained from a well-composed shape X with respect to a digital rigid transformation T is not necessarily well-composed itself. In that case, its topological properties are, of course, altered, and in particular, its boundary is no longer a (set of) 1-manifold(s). See Figure 4.1 for an example.

5. Rigid transformations and topology preservation

In this section, we summarize the main contribution of this work, that consists of defining a subfamily of well-composed shapes that remain stable – and topologically invariant – under any digital rigid transformations. The digital shapes forming this family are called *regular*, in reference to the above notion of *r*-regularity for continuous shapes in $\mathbb{R}^2$. The set of regular digital shapes can be defined in $\mathbb{Z}^2$ as follows.

Definition 5 ((Non-)singular shapes)**.** Let $X \subset \mathbb{Z}^2$ be a digital shape. We say that X is *singular* if at least one point $\mathbf{x}$ of X (resp. $\overline{X}$) has its whole 4-adjacent set included in $\overline{X}$ (resp. X).

Definition 6 (Regularity [7, 8])**.** Let $X \subset \mathbb{Z}^2$ be a non-singular, well-composed shape. We say that X is *regular* if for any 4-adjacent points $\mathbf{x}, \mathbf{y} \in X$ (resp. $\overline{X}$), there exists a 2×2 set $\boxplus = \{z, z+1\} \times \{t, t+1\} \subset \mathbb{Z}^2$ such that $\mathbf{x}, \mathbf{y} \in \boxplus \subseteq X$ (resp. $\overline{X}$).

The regularity of a digital shape can be characterized as follows, and thus leads to a linear-time complexity pattern-based regularity analysis.

Proposition 7 ([7, 8])**.** *A digital shape $X \subset \mathbb{Z}^2$ is regular iff none of the configurations depicted in Figure 5.1 (up to $\frac{\pi}{2}$ rotations, symmetries and value inversion) appears in X and $\overline{X}$.*

Note that the first configuration (Figure 5.1(a)) characterizes the non-well-composed shapes [5].

The main interest of regularity is to guarantee the stability of well-composed shapes – and their topological invariance – under any rigid transformation. In particular, we have the following result.

Theorem 8 ([8])**.** *Let $X \subset \mathbb{Z}^2$ be a well-composed shape. If X is regular, then, for any digital rigid transformation T, the transformed digital shape X_T is still well-composed, and has the same topological structure as X.*

Remark 9. This result establishes regularity as a sufficient condition for the stability of well-composedness, together with topological invariance. Our conjecture is however that this condition is also necessary.
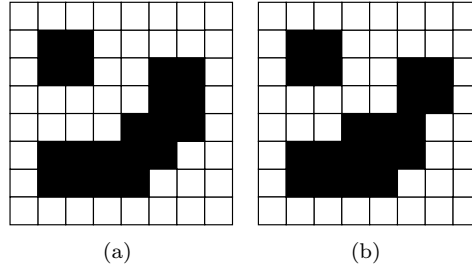
Figure 5.2: (a) A regular shape (in black). (b) A non-regular shape (in black) that is however opened and closed by a structuring element ⊞.

Following mathematical morphology terminology [6, Ch. 1], if X is regular, then X is open and closed by any structuring element ⊞ (see Definition 6), *i.e.*

$$\gamma_{\boxplus}(X) = X \ominus \boxplus \oplus \boxplus = X \tag{5.1}$$

$$\phi_{\boxplus}(X) = X \oplus \boxplus \ominus \boxplus = X \tag{5.2}$$

However, the converse is not true, as illustrated in Figure 5.2. Nevertheless, it is plain that there exist strong links between these morphological operations and the notions of regularity and topological invariance. Our conjecture is that the regular shapes X are exactly those whose both the dilated $X \oplus \boxplus$ and the eroded $X \ominus \boxplus$ have the same topological structure as X. This intuition derives from the continuous analogue of this assertion for r-regular shape S of $\mathbb{R}^2$, where the discs of radius r play the role of ⊞ [4].

6. CONCLUSION

This work constitutes a preliminary attempt to provide solutions for curvature definition and analysis of digital shapes, not only in their initial space but also in the spaces obtained under rigid transformations.

Some results have been proposed in the specific case of shapes in $\mathbb{Z}^2$ under rigid transformations. It has been proved in [8] that these results can, of course, be interpreted in the framework of binary digital images, but also extended to grey-level and label images, thus providing efficient image processing and analysis strategies.

Nevertheless, many efforts remain to do towards solutions in more general cases. In particular, the handling of (i) higher dimensions, *i.e.*, $\mathbb{Z}^3$ and more generally $\mathbb{Z}^n$ for $n \geq 3$, and (ii) arbitrary geometric transformations, still remain open issues. Moreover, it may be important to derive not only sufficient, but also necessary conditions, for curvature analysis. It may additionally be useful to deal with both Eulerian and Lagrangian models.

Even though this preliminary study allows us to understand and solve some topological problems of digital shapes under their rigid transformations, geometrical problems still remain: geometry of digital shapes is not preserved under rigid transformations in general, as exemplified in Figure 6.1. It is expected to investigate geometry-preserving conditions of digital shapes during their rigid transformations as well.
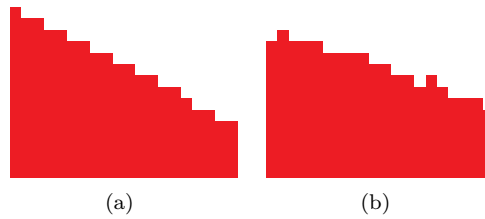
(a)   (b)

Figure 6.1: (a) A digital half plane, which is regular, and (b) its transformed shape, which is still well-composed (*i.e.*, the topology is preserved) but not a digital half plane any more (*i.e.*, the geometry is not preserved).

References

[1] A. Gross and L. Latecki. Digitizations preserving topological and differential geometric properties. *Computer Vision and Image Understanding*, 62(3):370–381, 1995.

[2] R. Klette and A. Rosenfeld. *Digital Geometry: Geometric Methods for Digital Picture Analysis*. Morgan Kaufmann, 2004.

[3] T. Y. Kong and A. Rosenfeld. Digital topology: Introduction and survey. *Computer Vision, Graphics, and Image Processing*, 48(3):357–393, 1989.

[4] L. J. Latecki, C. Conrad, and A. D. Gross. Preserving topology by a digitization process. *Journal of Mathematical Imaging and Vision*, 8(2):131–159, 1998.

[5] L. J. Latecki, U. Eckhardt, and A. Rosenfeld. Well-composed sets. *Computer Vision and Image Understanding*, 61(1):70–83, 1995.

[6] L. Najman and H. Talbot, editors. *Mathematical Morphology: From Theory to Applications*. ISTE/J. Wiley & Sons, 2010.

[7] P. Ngo, N. Passat, Y. Kenmochi, and H. Talbot. Well-composed images and rigid transformations. In *ICIP 2013, 20th International Conference on Image Processing, Proceedings*, pages 3035–3039, Melbourne, Australia, September 15–18 2013. IEEE Signal Processing Society.

[8] P. Ngo, N. Passat, Y. Kenmochi, and H. Talbot. Topology-preserving rigid transformation of 2D digital images. *IEEE Transactions on Image Processing*, 23(2):885–897, 2014.

[9] T. Pavlidis. *Algorithms for Graphics and Image Processing*. Springer-Verlag, 1982.

LIGM, Université Paris-Est, France • yukiko.kenmochi@esiee.fr

CEA LIST - DIGITEO Labs, France • ngo.diemphuc@gmail.com

CReSTIC, Université de Reims, France • nicolas.passat@univ-reims.fr

LIGM, Université Paris-Est, France • hugues.talbot@esiee.fr