

Bijective Maps from Simplicial Foliations

Marcel Campen*

Cláudio T. Silva
New York University

Denis Zorin

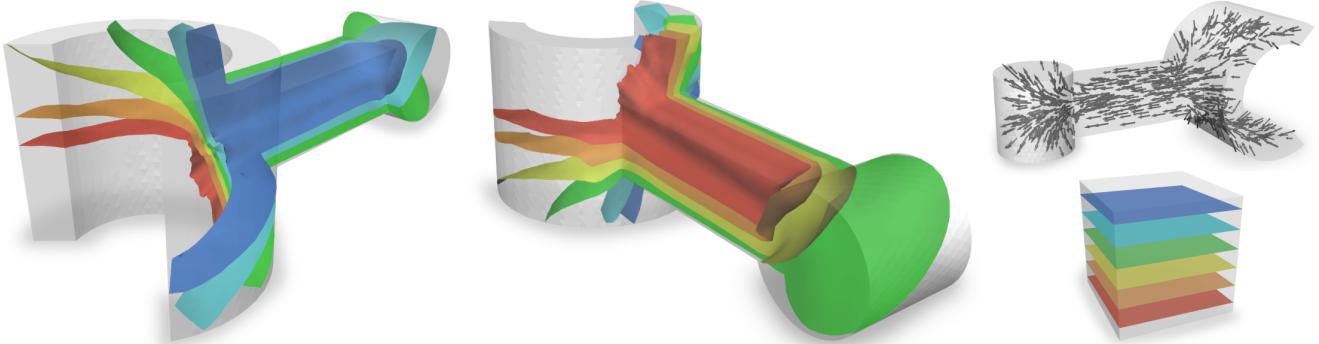


Figure 1: Volumetric map between an object and the unit cube, obtained with our approach. It is visualized with several isosurfaces. Bijectivity is given by construction, based on a combinatorially constrained volumetric vector field synthesis technique (upper right).

Abstract

This paper presents a method for bijective parametrization of 2D and 3D objects over canonical domains. While a range of solutions for the two-dimensional case are well-known, our method guarantees bijectivity of mappings also for a large, combinatorially-defined class of tetrahedral meshes (shellable meshes). The key concept in our method is the piecewise-linear (PL) foliation, decomposing the mesh into one-dimensional submanifolds and reducing the mapping problem to parametrization of a lower-dimensional manifold (a foliation section). The maps resulting from these foliations are proved to be bijective and continuous, and shown to have provably bijective PL approximations. We describe exact, numerically robust evaluation methods and demonstrate our implementation's capabilities on a large variety of meshes.

Keywords: parametrization, bijection, morphing, deformation, shelling, shape map, volumetric mapping

Concepts: •Computing methodologies → Mesh geometry models; Volumetric models;

1 Introduction

Parameterization, i.e., the mapping of shapes or their parts to simpler domains, is a fundamental operation in a broad range of applications involving complex geometry. Most scenarios require that the parametrization is a one-to-one map between the shape and its image in the parametric domain, as many downstream algorithms fundamentally rely on this essential bijectivity assumption.

* e-mail:campen@nyu.edu

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM.

SIGGRAPH '16 Technical Paper,, July 24 - 28, 2016, Anaheim, CA,
ISBN: 978-1-4503-4279-7/16/07
DOI: <http://dx.doi.org/10.1145/2897824.2925890>

The most important bijectivity result in the 2D setting is the Radó-Kneser-Choquet theorem [Kneser 1926; Choquet 1945] and its discrete variants [Tutte 1963; Floater 2003]. It assures that for a fixed bijective boundary map the (discrete) harmonic extension to the interior is bijective if the chosen boundary values bound a convex domain. Countless methods make use of this result. Even if a non-harmonic parametrization following other objectives is sought, it still is commonly exploited to robustly obtain a valid initialization.

In 3D, the problem is equally important: some example applications include morphing of volumes, transfer of interior structures or volumetric textures, template fitting, structured remeshing, compatible meshing via volumetric correspondences, cage-based deformations, and the mapping to regular grids to enable parallelization of volumetric computations.

However, the problem is qualitatively more difficult in 3D and higher dimensions: analogues of the Radó-Kneser-Choquet theorem and Tutte's theorem do not hold (harmonic maps and convex-combination maps to convex domains in 3D are *not* necessarily bijective, cf. Figure 2), and no other easily computable map is known that is guaranteed to exist and be bijective in general. The discretization of the 3D problem is also significantly more complicated. While in 2D a simple and easily realizable condition (3-connectedness) ensures that a mesh can be mapped to a convex domain piecewise linearly, no such conditions are known in 3D.

In this paper, we propose a method for the computation of bijective parametrizations, focusing on 2D and 3D domains. Specifically, we target the problem of parameterizing a simply-connected triangle mesh over the unit disk or square, or a tetrahedral mesh over the unit ball or cube, by constructing a bijective map between them. Maps between arbitrary domains can be derived from this fundamental operation by composition of these parametrizations and their inverses.

Contributions. The core concept of our construction is that of a *foliation*: the domain is equipped with a structure partitioning it into disjoint one-dimensional *leaves*. This makes it possible to reduce the problem to a lower-dimensional problem of parameterizing a *transversal section* of the original domain, and one-dimensional curve parametrization problems, both of which are easily solved.

The combination of the lower-dimensional parametrizations in a tensor product manner yields a bijective parametrization, i.e. a func-

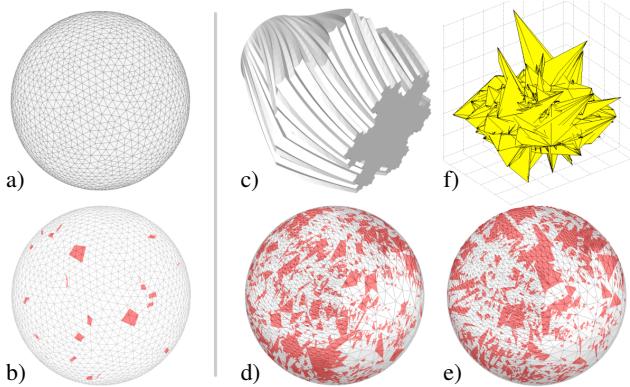


Figure 2: Harmonic and Tutte maps are not bijective in 3D. *a)* Tetrahedral mesh of a unit ball. *b)* The mesh of (*a*) mapped using the 3D analog of Tutte’s embedding (with uniform weights). The boundary was fixed using the identity map on the sphere, i.e. this is a maximally benign setup. Still, the map is not bijective: several internal tetrahedra are inverted (highlighted in red). *c)* Tetrahedral mesh of a more complex shape. *d)* This mesh Tutte-mapped and *e)* discrete harmonically (cotan-Laplacian) mapped to the unit ball. Several thousand internal tetrahedra are inverted. The non-injectivity can be so severe that even the powerful recent map improvement code based on bounded-distortion projection [Aigerman and Lipman 2013] is unable to repair it: regardless of its parameter setting ($K \in \{20, 50, 100, 200, 500, 1000, 2000\}$) the issue only gets worse (*f*, inverted tetrahedra in yellow).

tion that, given a point in the object domain, returns the unique corresponding point in parameter space. The *inverse*, i.e. a function that, given a point in parameter space, returns the unique corresponding point in the object, is immediately available as well.

This general principle allows to bijectively parameterize any simply-connected domain (in any dimension). We treat herein the 2D and 3D cases, with a triangle mesh or tetrahedral mesh discretization of the domain, and present a practical solution based on *PL foliations*, with leaves which are linear per mesh simplex. This kind of discretization restricts applicability to *shellable* meshes, and the efficient algorithm we introduce is guaranteed to succeed if the mesh is *extendably shellable*. We discuss these combinatorial properties and their theoretical implications in detail, and present empirical evidence that this restriction is likely of low practical relevance.

We show that, by appropriate algorithmic and representational choices, the method can be implemented in a *numerically robust* manner, such that bijectivity is not only a theoretical property of our approach.

Piecewise linear representations or approximations of the above pointwise defined maps are of interest as well. As the input mesh is not generally suited for this purpose, we discuss how a bijective PL representation can be obtained via refinement.

Besides bijectivity, the *quality* of the parametrization in terms of incurred distortion is relevant. We describe complementary strategies to control quality throughout our construction.

Outline. After reviewing previous work and discussing the basic concepts related to foliations in Section 3, we present and analyze an algorithm to construct such foliations in a discrete setting in Section 4. The robust computation of bijective maps from these foliations is detailed in Section 5. In Section 6 we discuss how the degrees of freedom of the method can be exploited to control the quality of the map within the space of bijections. In Section 7, towards enabling a broader range of quality optimization methods, we show how the resulting bijective map can be approximated piecewise linearly, with a guarantee that the resulting PL map is bijective.

2 Related Work

Harmonic Parametrization. Prescribing a homeomorphism between the boundary of a simply-connected two-manifold and the unit circle, one can consider the solution of the corresponding Dirichlet problem: a harmonic map, known to be a diffeomorphism to the unit disk [Choquet 1945]. For certain discretizations of the Laplacian (with non-negative weights) the resulting piecewise-linear convex-combination maps on triangle meshes can be shown to always be injective as well [Floater 1997; Floater 2003]. Even the most recent parametrization methods still rely on this result in various ways for bijectivity in 2D [Weber and Zorin 2014; Myles et al. 2014; Smith and Schaefer 2015; Schneider and Hormann 2015; Aigerman et al. 2015; Aigerman and Lipman 2015].

This nice property does not carry over to the three-dimensional volumetric case, neither in the continuous setting (explicit counterexamples provided in [Laugesen 1996; Liu and Liao 1996]) nor in the discrete, simplicial mesh setting [Colin de Verdière et al. 2003; Floater and Pham-Trong 2006], cf. Figure 2. While discrete harmonic maps or fields were used in a few instances to produce bijective maps, [Alexa et al. 2000; Wang et al. 2004; Li et al. 2007; Lin et al. 2015], they are not guaranteed to be bijective in general. In fact, bijectivity of harmonic maps in 3D “fails badly” [Laugesen 1996] in the following sense: any boundary homeomorphism can be turned, by an arbitrarily small perturbation, into one whose harmonic extension is non-injective – even if both target and source shapes under consideration are as nice and simple as a ball.

Harmonic Gradient Parametrization. Based on the physical analogy of an electrostatic potential, a number of authors have constructed parametrizations from gradient lines of a harmonic function. Our method is most closely related to this set of ideas. The two main approaches of this type involve either one internal critical point (a point charge, inducing a Green’s function) or constant potentials on subsets of the boundary. For a point charge, injectivity is easily shown in the 2D setting, but it is long known that generalization to 3D fails [Gergen 1930], except for special cases like convex or star-shaped objects [Gergen 1931; Xia et al. 2010a]. The use of boundary potentials [Xia et al. 2010b; Nguyen and Jüttler 2010] does not guarantee bijectivity in 3D either [Stoddart et al. 1964].

A promising related idea is to replace the Green’s function with one specifically tailored to be free of additional critical points even if the object is not star-shaped. Huynh and Gingold [2015] outline such an approach (relying on a regular grid instead of a mesh), which poses interesting challenges related to boundary conformance and correct definition and practical tracing of the function’s gradient in a smooth manner, as required for injectivity.

Constrained Deformation and Optimization. A parameterization over a specific domain like the unit ball or cube can potentially be generated via orientation-preserving, non-degenerate deformation of the original mesh to the target domain. While deformation techniques are available [Schüller et al. 2013; Jin et al. 2015], preserving injectivity by constraints or barrier functions [Hormann and Greiner 2000; Degener et al. 2003; Fu et al. 2015], they cannot guarantee that the deformation actually reaches the desired state, matching the target domain. On the other hand, optimization methods that produce provably injective mappings using convexified constraints and starting from a non-bijective initial point possibly already satisfying boundary constraints [Aigerman and Lipman 2013; Kovalsky et al. 2014] cannot guarantee that a solution to this specific problem instance is found. One of the inherent issues is the fact that the parametrizations induced by such deformations are linear per simplex of the original mesh, and the tessellation or resolution of the mesh might preclude bijectivity on the way to or at the target configuration [Jin et al. 2014; Weber and Zorin 2014]. Remeshing or refinement can be mandatory, particularly in 3D.

We note that the success of such approaches depends on a number

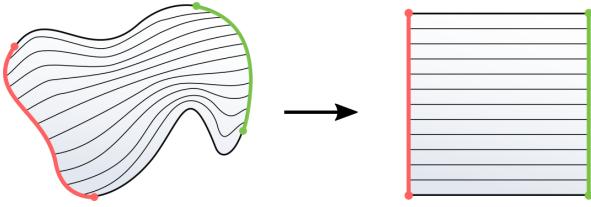


Figure 3: A foliation of a simply-connected 2-manifold (with boundary) and a compatible foliation of the unit square. The black curves show some of the foliations' leaves. Two transversal sections of the foliation are shown in red and green.

of factors, including mesh connectivity, mesh geometry, boundary map, and non-linear optimization strategy. The property of mesh shellability our algorithm relies on depends purely on connectivity.

Theory of PL Maps. The non-trivial fact that the interior of any 3D polyhedral domain can actually be mapped piecewise linearly to a simple domain (e.g., a simplex) was proved by Moise [1952], using ideas from Alexander [1924]. Moise also shows that a common refinement exists for any two tetrahedralizations of the same manifold (solving the Hauptvermutung for 3D). Later, Pachner [1991] has shown that any two homeomorphic tetrahedralizations are related by sequences of elementary connectivity operations called (*inverse*) *shellings* (similar to edge collapses in triangle meshes). Regarding the amount of mesh refinement needed to be able to define a PL bijection between tetrahedral meshes, upper bounds have been established [King 2004].

Several positive results related to shellings were established by Mani and Bruggesser [1971] and subsequent papers, in particular *shellability* of convex polytopes and, as a consequence, Delaunay tetrahedralizations. Surveys of related results can be found in [Ziegler 1998] and [Adiprasito and Benedetti 2012].

Foliation Constructions. Foliations are closely related to vector fields without intersecting (merging/splitting) integral curves. For the 2D case, some constructions for such vector fields on triangle meshes have been described, involving non-linear interpolation [Zhang et al. 2006]. Using careful approximation techniques [Ray and Sokolov 2014; Myles et al. 2014], integral curves of such fields, implying a foliation in 2D, can be traced piecewise linearly.

3 Foliations

3.1 Idea and Background

Informally, a *foliation* with one-dimensional *leaves* is a decomposition of a domain M into disjoint curves (Figure 3 left) [Moerdijk and Mrčun 2003]. The idea our method builds upon is to create a foliation topologically equivalent to the canonical Cartesian foliation of the unit square (Figure 3 right), i.e. there is a one-to-one continuous correspondence between leaves having the same topology. This allows construction of a bijective map between M and the square from a lower-dimensional bijective map on the *section* of M depicted in red and one-dimensional bijective maps of individual leaves (e.g. to the interval $[0, 1]$). The lower-dimensional map of the section establishes a one-to-one correspondence between leaves. As all leaves are disjoint, and paired-up leaves are mapped to each other bijectively, the overall map is bijective.

Other topological types of foliation can be employed as well: e.g., a bijective map between M and the (punctured) unit disk can be constructed using a foliation topologically equivalent to the disk's canonical polar foliation, cf. Figure 4. This can be seen as a generalized form of the point charge field lines mapping [Gergen 1930; Xia et al. 2010a; Huynh and Gingold 2015].

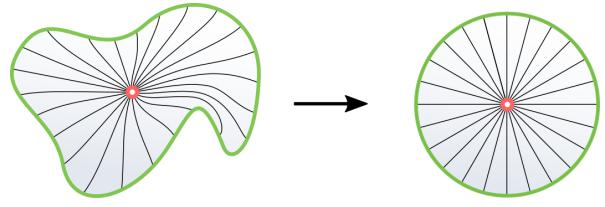
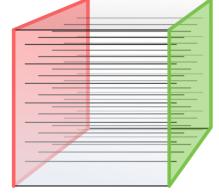


Figure 4: Mapping using a polar foliation of the punctured disk.

In 3D, sections of the foliation are two-dimensional, thus a two-dimensional section map is required to establish correspondences of leaves (a problem that can be recursively reduced to one-dimensional parametrization again). With this section map, in a manner analogous to the 2D case, bijective maps to the cube or the ball can be established. Next, we introduce more precise definitions to set the stage for the description of our algorithm.



Definitions and Notation

We construct maps from a closed domain M (a triangle or a tetrahedral mesh) homeomorphic to an n -ball, $n = 2$ or 3, to a standard domain D (the unit square, disk, cube, or ball).

Definition 1. A *foliation* is a decomposition of a manifold M into disjoint low-dimensional submanifolds (*leaves*) of constant dimension (we only need to consider dimension 1).

For the purpose of this paper, a very specific, simple type of foliations, which we call *trivial*, is sufficient: the leaves are 1-dimensional and each one is homeomorphic to a line segment, i.e., there are no closed or infinite leaves. A more general type of foliations may be useful for other types of domains (and parts of our method are applicable also to non-ball domains as well as in dimensions higher than 3).

A *singular foliation* may contain singular leaves of lower dimension. We consider only one type of singular foliations: *radial foliations* that have one point singularity, the center, that is in the closure of all leaves. To simplify exposition, we consider instead nonsingular foliations of the domain obtained by removing a small open neighborhood (a single tetrahedron or a triangle) in which the singularity is located. Maps constructed using this reduced domain are easily extended into the removed simplex.

A *transversal section* \mathcal{S} of a foliation with 1-dimensional leaves is a submanifold of dimension $n - 1$ that intersects every leaf exactly once. For instance, in Figures 3 and 4 the red and green curves are sections of the depicted foliations. Assuming that each leaf is parametrized by $t \in [0, 1]$, the transversal sections that intersect all leaves at $t = 0$ or $t = 1$ is called the *source section* (red curve) and the *sink section* (green curve), respectively.

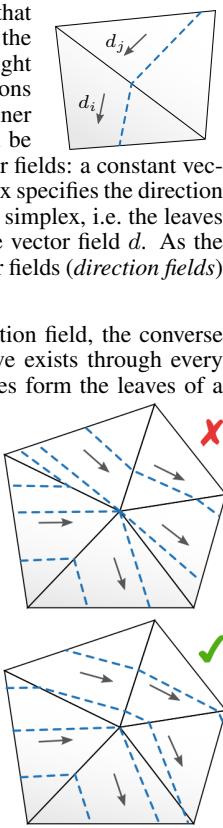
Maps from Foliations

Given a trivial foliation of the n -manifold M , any point $p \in M$ can now uniquely be identified by a pair (s_p, t_p) where s_p is the point on the section \mathcal{S} intersected by the leaf running through p , and t_p is the parameter of p along this leaf.

Assume we know a bijective continuous map ψ between \mathcal{S} and $[0, 1]^{n-1}$ or S^{n-1} , which are sections of the canonical Cartesian or polar foliations of D (red or green in Figures 3 and 4 right), respectively. Then $\Psi(p) = (\psi(s_p), t_p)$ is a bijective map from M to $D = [0, 1]^n$ or $D = S^{n-1} \times [0, 1]$. This map effectively equips M with Cartesian or polar coordinates, respectively, providing a parametrization over the unit square/cube or disk/ball. With an appropriate choice of t (e.g. normalized arclength) Ψ is continuous.

PL Foliations and Direction Fields. If M is a triangulated domain of dimension $n = 2, 3$, consisting of triangles or tetrahedra, a natural class of foliations to consider is that of *piecewise-linear foliations*, for which the leaves inside each simplex are parallel straight line segments. Such piecewise-linear foliations can be handled in a numerically robust manner more easily than other types, and they can be described by piecewise-constant (PC) vector fields: a constant vector d_i assigned to the interior of an n -simplex specifies the direction of the parallel straight leaf segments in this simplex, i.e. the leaves are the integral curves (dashed blue) of the vector field d . As the vector's magnitude has no effect, unit vector fields (*direction fields*) are appropriate.

While any PL foliation implies a PC direction field, the converse is not true. Only if a unique integral curve exists through every point in M do these piecewise-linear curves form the leaves of a foliation (which are disjoint by definition). While this property of unique integral curves is well known, e.g., for the class of uniformly Lipschitz continuous vector fields, it does not generally hold for discontinuous fields, like the PC fields in question: integral curves may merge and split (as depicted on the right, top), unlike the leaves of a foliation (bottom). This difference is crucial, and it is one of the main obstacles for using PC fields for parametrization directly. Below, we derive a combinatorial condition on these fields that ensures correspondence to a foliation. We call a piecewise-constant field that has a unique integral curve passing through each point *foliation-compatible*.



3.2 Generation Algorithm

We introduce a basic algorithm to construct such foliation-compatible direction fields subject to constraints regarding the behavior at the domain boundary and the absence of singularities. In Section 4.4 we analyze and discuss it in detail.

We consider a simplicial mesh M , consisting of k -simplices, $0 \leq k \leq n$, with $n = 2$ (triangle mesh) or $n = 3$ (tetrahedral mesh). For an n -simplex c and an incident $(n-1)$ -simplex f , the normal of f pointing outside of c is denoted n_{cf} . We assume that no n -simplex has multiple boundary $(n-1)$ -simplices.

- 1: **Input:** Mesh M , with a simply connected subset of boundary $(n-1)$ -simplices marked *source*, and another subset marked *sink*. The remaining boundary $(n-1)$ -simplices are marked *parallel*, and the interior is unmarked. Set $\bar{M} = \emptyset$.
- 2: **while** $M \neq \emptyset$ **do**
- 3: Select a *free* n -simplex $c \in M$ with one or more *source* faces.
- 4: Choose a direction d_c such that for all incident $(n-1)$ -simplices f :
- 5: • $d_c \cdot n_{cf} < 0$, if f marked *source*
- 6: • $d_c \cdot n_{cf} > 0$, if f marked *sink* or unmarked
- 7: • $d_c \cdot n_{cf} = 0$, if f marked *parallel*
- 8: $M = M \setminus c$, mark the $(n-1)$ -simplices that are revealed *source*, $\bar{M} = \bar{M} \cup \{c\}$
- 9: **Output:** Foliation-compatible direction field d .

Algorithm 1

An n -simplex is *free* if removing it from M and adding it to the complement \bar{M} leaves both, M and \bar{M} , manifold. This is not the case iff a singular vertex or edge (whose immediate neighborhood does not have ball or half-ball topology) is created in M or \bar{M} . One may ask whether such a free simplex is always available. As we shall see in Section 4.4, a necessary condition for this is that the mesh is (bi)shellable, a sufficient condition is that it is extendably (bi)shellable. In 2D this is no hurdle, in 3D there are known

counterexample constructions; their practical relevance appears to be insignificant in our tests, as we show in more detail below.

Practical questions, particularly concerning the choice of c and the choice of d_c , will be made concrete later; for analysis purposes this abstract form of the algorithm suffices.

Next, we discuss why the resulting field is guaranteed to be foliation-compatible and to imply a trivial foliation. In Section 5 we explain how to construct an explicit map $\Psi : M \rightarrow D$ and its inverse from the foliation-compatible field.

4 Foliation Compatibility Criteria

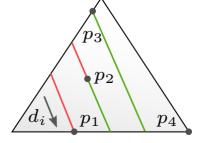
For a PC direction field to represent a foliation, a *local* condition on the field needs to be satisfied. For it to represent a *trivial* foliation, an additional *global* condition is needed. Importantly, both local and global conditions can be formulated purely in terms of combinatorics of the discrete direction field.

4.1 Local Condition

To simplify exposition, we initially treat *generic* fields, i.e. the direction d of an n -simplex is not parallel to any of its edges or faces in the interior.

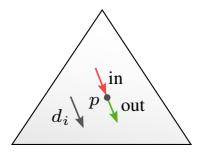
Incoming and outgoing simplices. As a first step in formulating our combinatorial requirement, consider an arbitrary point p in M ; the basic requirement is that there is exactly one integral curve of the field d passing through this point. Consider a field direction d_i on an n -simplex c_i which contains point p (in its interior or on its boundary), and the segment ℓ of the line passing through p in direction d_i , restricted to c_i . Removing the point p from ℓ splits ℓ into two parts, ℓ_- and ℓ_+ corresponding to directions $-d_i$ and d_i .

Definition 2. (Incoming/outgoing n -simplex) If the half-segment ℓ_- is non-empty, we call the simplex c_i *incoming* for p (it provides an incoming integral curve for p), and if ℓ_+ is non-empty, c_i is *outgoing* for p . In the example on the right, where ℓ_- is red, ℓ_+ green: c_i is incoming for p_1 and p_2 , outgoing for p_2 and p_3 , and neither for p_4 .

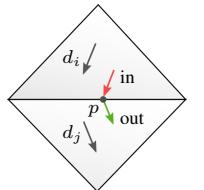


Due to the piecewise constant nature of the field, if an n -simplex c is incoming (outgoing) for a point p in the relative interior of a k -simplex f , $k < n$, it is incoming (outgoing) for every point in the interior of f . Hence, we can say that an n -simplex c is incoming (outgoing) for a k -simplex f .

As we assume here that the field is generic, for any point in the interior of M there needs to be exactly one incoming and one outgoing n -simplex. If p is contained in the interior of an n -simplex c , then this requirement is satisfied automatically: c is incoming and outgoing for p . For a point p in the relative interior of a k -simplex, $k < n$, we need to consider its incident n -simplices:



For a point p on an $(n-1)$ -simplex f incident to n -simplices c_i and c_j , we must require that c_i is incoming for p (equivalently: f) and c_j outgoing, or vice versa, as each incident n -simplex can only be either, incoming or outgoing for f . For a k -simplex, $k < n-1$, the situation is more complex as there are multiple incident n -simplices. These observations immediately lead us to the following proposition:



Proposition 1. A generic PC direction field is foliation-compatible if and only if each interior k -simplex, $k < n$, has a unique incoming

and a unique outgoing n -simplex; each boundary k -simplex has at most one incoming and one outgoing n -simplex.

Note that this is *not* sufficient for the foliation to define a *trivial* foliation, as integral lines may still be infinite or closed.

Extended complex. To unify and simplify notation in the following (avoiding special cases on the source and sink sections), we define the complex M^* . It is obtained by adding two n -cells, c_{source} and c_{sink} to the complex M . These cells are *virtual*, they have no geometric realization, are not simplicial, and not necessarily of disk-ball topology. They are connected to M such that they cover exactly the source and sink section, respectively, i.e. M^* 's boundary simplices in the source and sink region are their lower-dimensional faces. These virtual cells are defined to implicitly be incoming (c_{source}) and outgoing (c_{sink}) for their faces from now on.

We extend the notion of generic fields to *generic boundary-aligned* fields, i.e. the directions in boundary n -simplices are parallel to the adjacent boundary $(n-1)$ -simplices. Note that in the case of M^* this affects exactly the simplices which are marked *parallel* (the rest of M 's boundary is covered by the virtual cells in M^*). As integral curves cannot start or end at the boundary where the field is parallel, this leads to the following simpler variant of Proposition 1:

Corollary 1. A generic boundary-aligned PC direction field is foliation-compatible if and only if each k -simplex, $k < n$, has a unique incoming and a unique outgoing n -simplex.

Dual graph. To formulate our combinatorial condition, we need to consider the dual edge graph $G(M^*)$ of the complex M^* , i.e., the graph with vertices corresponding to n -cells of M^* , and edges connecting n -cells sharing an $(n-1)$ -simplex. Note that the condition that for each interior $(n-1)$ -simplex f one of the two incident n -simplices c_i and c_j is incoming and the other is outgoing, allows for two choices that can be modeled by assigning an orientation to the dual edge connecting c_i and c_j . We use $c_i \rightarrow c_j$ to denote that c_i is incoming for f and c_j is outgoing. Let O be an assignment of an orientation to each edge, then we denote by $G(M^*, O)$ (short: G) the corresponding *directed* dual edge graph.

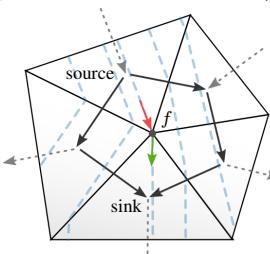
Definition 3. (Local bipolarity) G is called *locally bipolar* if for each k -simplex f , $k < n$, its restriction G_f to f 's star (i.e., to the set of n -simplices incident at f), has exactly one local source (vertex with indegree 0) and one local sink (vertex with outdegree 0).

For an $(n-1)$ -simplex f this holds trivially: the star consists of two n -simplices and the restriction G_f has a single directed edge (or, at the boundary, a single vertex, which is source and sink). For lower-dimensional simplices (in 2D, vertices, in 3D, vertices and edges), restrictions of G may (as depicted on the right for a vertex) or may not be bipolar.

We can now formulate our local criterion, which we will use to demonstrate that the output of Algorithm 1 is foliation-compatible. This condition is necessary and sufficient, i.e., foliation compatibility of a PC field is entirely a combinatorial fact.

Let O_d be the (partial) orientation implied by a field d : $c_i \rightarrow c_j$ if c_i is incoming for, and c_j is outgoing for the common $(n-1)$ -simplex. This implied orientation may be partial: if c_i and c_j are both incoming or outgoing, the corresponding edge remains undirected. Partially oriented graphs never are locally bipolar.

Proposition 2. A generic boundary-aligned PC direction field d on a complex M^* is foliation-compatible, if and only if the corresponding oriented dual edge graph $G(M^*, O_d)$ is locally bipolar (see Appendix A.1 for a proof).



4.2 Global Condition

While the local condition guarantees that there is a unique leaf passing through each point, globally the leaves may not have the desired behavior: a leaf may be closed or infinite, without ever reaching a source or a sink section, i.e. the implied foliation may be nontrivial. While foliation compatibility is a purely combinatorial notion, trivial foliation compatibility is not: two fields d and d' with the same directed graph ($O_d = O_{d'}$) may have different global topology. However, one can formulate a *sufficient* combinatorial condition for trivial foliation-compatibility:

Definition 4. (Global bipolarity) A digraph G is called *globally bipolar* if it is acyclic and has exactly one source and one sink.

Note that we are particularly interested in orientations where c_{source} and c_{sink} are the global source and sink node of the graph.

Proposition 3. If $G(M^*, O_d)$ is locally and globally bipolar, the field d is compatible with a trivial foliation.

Indeed, consider a leaf of the foliation and construct a corresponding path q in the directed graph $G(M^*, O_d)$ as follows: if the leaf passes from one n -simplex to another through an $n-1$ -simplex, the corresponding dual edge is included in q . If it passes through a k -simplex f for $k < n-1$, it has to pass through the local source and sink nodes of G_f . As G_f is connected and bipolar, there exists at least one path of dual edges in G_f from local source to sink, which we add to q . This dual path q cannot terminate anywhere except at a source or a sink (otherwise it could be extended), so it has to either connect a source with a sink, or be a cycle. As the graph is acyclic by assumption, all leaves connect source and sink section.

By Propositions 2 and 3, the question of trivial foliation compatibility is reduced to combinatorial conditions. In the following we show that Algorithm 1 respects these conditions.

4.3 Shelling

For the analysis of the Algorithm we will make use of the concept of *shelling*. Let m denote the number of n -simplices in M .

Definition 5. (Shelling) An ordering c_1, c_2, \dots, c_m of the n -simplices is called a *shelling* or *shelling order* of an n -manifold, if every suffix of the order is manifold as well [Dey et al. 1999].

In the literature there is a number of similar definitions; in some cases this order is defined in reverse sense, such that every prefix is manifold instead; we call this a *reverse shelling*.

Definition 6. (Bishelling) An ordering c_1, c_2, \dots, c_m that is a shelling as well as a reverse shelling we call *bi-directional* shelling, or short: *bishelling*. The notation $c_1 c_m$ -*bishelling* is used to refer to an ordering with specific first and last elements c_1 and c_m .

An ordering of the n -simplices (whether shelling, bishelling, or neither) implies an orientation for the dual graph G in an obvious manner: the dual edge between c_i and c_j is oriented $c_i \rightarrow c_j$ iff $i < j$.

Proposition 4. The orientation of the dual graph G implied by a $c_1 c_m$ -*bishelling* is both locally and globally bipolar, and c_1, c_m are the global source and sink (see Appendix A.2 for a proof).

4.4 Analysis of the Algorithm

Algorithm 1, by greedily selecting free n -simplices one-by-one, effectively constructs a bishelling order. This is due to the restriction to *free* n -simplices, preserving the manifoldness of prefix and suffix in every iteration. More precisely, we conceptually apply the algorithm to M^* , starting with the virtual cell c_{source} . However, M^*

does not need to be constructed explicitly: the algorithm can actually be applied to M ; only the determination of freeness needs to take the existence of the virtual cells into account.

The bishelling constructed by Algorithm 1 (if it succeeds) then specifically is a $c_{\text{source}} c_{\text{sink}}$ -bishelling.

Note that the way directions are set in Algorithm 1 matches this orientation: at the moment n -simplex c_j is processed, a face separating it from c_i is *source* iff $i < j$ (c_i has been removed earlier) and is unmarked or *sink* iff $i > j$ (c_i is yet to be removed in a later step, or it is the final c_{sink}), so the *geometric* directions of d match the *combinatorial* orientation in G . In other words, the orientation O_d implied by the field d equals the orientation implied by the constructed $c_{\text{source}} c_{\text{sink}}$ -bishelling. It follows from Proposition 4 that d is compatible with a trivial foliation, as required for our purpose.

Bishellability

As mentioned before, not every mesh possesses a (bi)shelling order. The restrictions imposed by this are quite different in the 2D and 3D cases. We review here several known theoretical results, and present an extension of one, to address the *bishellability* of meshes, i.e. the existence of bishellings, in 2D and 3D.

2D case. In the 2D case, the dual graph of a 2-simplicial complex (triangle mesh) is planar. For planar graph orientations, local bipolarity follows from global bipolarity [de Fraysseix et al. 1995]. It is known that a global bipolar orientation, and equivalently a bishelling (also known as *st*-numbering in this case), exists if the graph is 2-connected [Lempel et al. 1967]. This is a very mild condition: if an input mesh violates it, subdividing all internal edges incident to two boundary vertices achieves 2-connectedness (of G). Afterwards, the mesh is shellable and bishellable.

3D case. In the 3D case, dealing with tetrahedral meshes, the theoretical situation is significantly different, although our experiments indicate that this difference is less significant in practice. In analogy to the 2D case, we first subdivide internal 1- and 2-simplices incident to only boundary vertices (ensuring that no tetrahedron has more than one boundary face), but this is not necessarily sufficient.

We first consider a helpful equivalence that holds in the case of radial foliations:

Proposition 5. Suppose c_1, \dots, c_m is a shelling of a simply-connected n -manifold M . Then c_0, c_1, \dots, c_m is a bishelling of the extended complex obtained by adding a virtual n -cell c_0 that covers the entire boundary of M (see Appendix A.3 for a proof).

This shows that in the case of a radial foliation the notions of shelling and bishelling are equivalent for our purpose: if we know a shelling of M , it is also a bishelling (of M^* , with $c_{\text{source}} = c_0$ and c_{sink} chosen as c_m). Thus a shelling is compatible with a radial foliation. This is helpful because shellings, in contrast to bishellings, were already extensively studied in the literature, and a number of theorems on shellability provide necessary conditions for the existence of a shelling.

Tetrahedral complexes with no shelling order do exist [Furch 1924; Rudin 1958]. Moreover, the worst-case complexity of finding a shelling for a shellable mesh is unknown. At the same time, it is known that important classes of meshes (e.g. Delaunay tetrahedralizations of 3D point sets) are shellable, and a shelling order can be found efficiently [Mani and Bruggesser 1971]. More generally, it was established that for any mesh there is an r such that after r barycentric refinements the mesh becomes shellable [Moise 1952]. For convex objects, r is bounded by 1 [Adiprasito and Benedetti 2012], for non-convex objects so far no polynomial bounds seem to be known [Adiprasito and Izmostev 2015], though the examples known to require $r > 1$ are intricate constructions based on compositions of exponentially many knotted tunnels [Goodrick 1968].

In the case of cube foliations, the notion of bishellability is stronger than that of shellability. While results on bishellings are less common, the results on shellings often can be extended. In particular, the following proposition can be proved:

Proposition 6. There is a refinement of the extended complex M^* that is $c_{\text{source}} c_{\text{sink}}$ -bishellable (see Appendix A.4 for a proof).

Most of these existence results are primarily of conceptual interest but do not constructively yield a bishelling order. In practice, one can attempt to find an order in a greedy manner in linear $O(m)$ time. This is what Algorithm 1 does. In 3D, it is thus not guaranteed to find a (bi)shelling order even if the mesh permits one—unless the mesh is *extendably* (bi)shellable [Ziegler 1998], which means that every partial (bi)shelling can always be extended to a complete one, i.e. no backtracking or look-ahead planning is necessary, thus any greedy strategy succeeds. One can further extend the algorithm, e.g. by repeatedly performing orderings with randomized free simplex selection order until success, but in our tests this proved to be entirely unnecessary, as the simple greedy variant succeeded in all but specifically constructed cases. This is detailed in the following.

Experimental Evaluation of Shellability. We applied Algorithm 1 to a set of 767 tetrahedral meshes; the primary sources of surface meshes were Thingiverse, from which 600 meshes were obtained, and the AIM@SHAPE repository. The Thingiverse meshes are primarily designed manually, while AIM@SHAPE meshes are mostly scanned. The tetrahedral meshes were generated using Tetgen, Netgen, or CGAL’s Delaunay mesher. A number of tetrahedral meshes of unknown origin were included. Some examples of Thingiverse dataset meshes are shown in Figure 5.

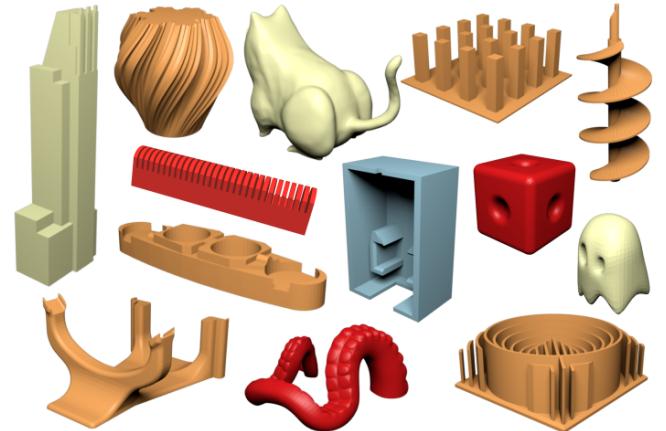
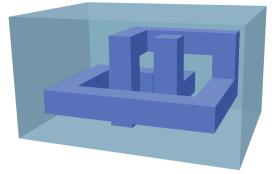


Figure 5: Some examples of meshes in our test data set. The meshes have a variety of protrusions, indentations, thin walls, and other types of features. Our algorithm robustly handles all of them.

For the case of a ball map the algorithm succeeded in obtaining a shelling order in all cases. For the case of a cube map foliation, where it seeks to construct a bishelling order, it succeeded on all but 15 input meshes, which had thin parts with many edges extending through the interior from boundary to boundary. However, a bishelling order was found for these meshes after one barycentric refinement step. We had to construct a mesh following a specific counterexample found in the literature, Furch’s ball [Furch 1924], to obtain a case where, even after two steps of barycentric refinement, the greedy Algorithm 1 could not find a bishelling order. The boundary of this object is depicted on the right, with transparent outer boundary to reveal the knotted dead-end tunnel entering the volume from the bottom.



5 Evaluating Bijective Maps

Once a foliation-compatible field d is constructed, for a given point $p \in M$ we can obtain its foliation coordinates to evaluate the map Ψ . Let \mathcal{S} be the source section of M and ψ is a bijective continuous map from \mathcal{S} to the corresponding source section of D . Let s_p be the point where the leaf ℓ_p , on which p is located, intersects \mathcal{S} , and $t_p \in [0, 1]$ the parameter of p along ℓ_p . Then t_p is called the *minor coordinate*, and the image $\psi(s_p)$ of the section point s_p is called *major coordinates*. Interpreted as Cartesian or polar coordinates, they define the map $\Psi(p) = (\psi(s_p), t_p)$. These coordinates are determined simultaneously by a relatively straightforward process of tracing a PC direction field through the mesh. Due to the constant nature per simplex, this amounts to a sequence of d -parallel projections through simplices.

5.1 Barycentric Field Representation

At each step, Algorithm 1 has to choose a direction d_i on an n -simplex c_i , pointing into c_i through certain incident faces and out of c_i through others. We represent the direction d_i as an offset vector in barycentric coordinates, i.e. (u, v, w) with $u + v + w = 0$ in the 2D case and (u, v, w, x) with $u + v + w + x = 0$ in the 3D case. Since these components can be interpreted as coefficients of the incident face (inward-pointing) normals, their signs directly determine the signs of the dot products of d_i with the outward normals n_{cf} of the $(n-1)$ -simplices. Thus, for a given incoming/outgoing configuration, we can arbitrarily choose non-zero absolute values for the components of $d_i = (u, v, w)$ or (u, v, w, x) , because the fixed signs already restrict the directional space to exactly those directions that fulfill the incoming/outgoing constraints. By construction, there is always at least one incoming and one outgoing incident face. This means that there are always components of both signs, and choices of (u, v, w) and (u, v, w, x) satisfying the constraints always exist.

This barycentric representation is not only convenient, it will also allow us to easily perform the subsequent steps in a numerically robust manner, as detailed in the following.

5.2 Leaf Tracing

We consider the 3D case; the 2D case (depicted in Figure 6) follows by reduction. Given a point p , represented in barycentric coordinates $(\alpha, \beta, \gamma, \delta)$ in tetrahedron c_i , where $d_i = (u, v, w, x)$.

The (supporting line of the) integral curve through p in c_i is given by the line equation $\ell_{p,c_i}(\lambda) = p - \lambda d_i = (\alpha - \lambda u, \beta - \lambda v, \gamma - \lambda w, \delta - \lambda x)$ with $\lambda \in \mathbb{R}$. The intersections of ℓ with the supporting planes of c_i 's triangular faces are easily determined based on the observation that one barycentric coordinate component vanishes on these planes. For instance, one of the four intersections is

$$\ell\left(\frac{\alpha}{u}\right) = \left(0, \beta - \alpha \frac{v}{u}, \gamma - \alpha \frac{w}{u}, \delta - \alpha \frac{x}{u}\right).$$

Note that some of the four intersection points can be equal (if ℓ intersects a vertex or an edge) or lie outside of the tetrahedron (if ℓ intersects the supporting plane but not the face).

This elementary operation allows us to trace the piecewise linear integral curve from one simplex to another. As barycentric coordinates are local, a change of coordinates is needed when crossing to the next n -simplex. As for points on a common k -simplex f the barycentric coordinates w.r.t. both incident n -simplices involve only the vertices of f , the transform amounts to simply permuting the barycentric coordinates according to the change in local vertex indices from one n -simplex to the other.

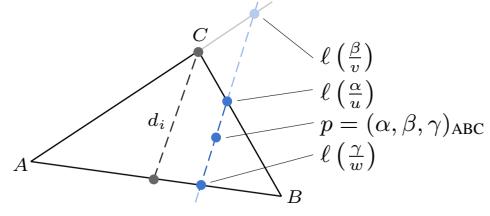


Figure 6: Direction field tracing through a triangle via parallel projection in barycentric coordinates.

5.3 Major Coordinates

From the point s where a leaf hits the section \mathcal{S} , the major coordinates are determined through ψ . For the different cases, ψ is obtained as follows:

Unit Square/Disk. If M shall be mapped to the square, $\psi : \mathcal{S} \rightarrow [0, 1]^2$. This is a simple piecewise linear curve parametrization problem, which can, for instance, be performed according to arc-length, normalized to the unit interval. If M is mapped to the disk, $\psi : \mathcal{S} \rightarrow S^1$, the unit circle, and a simple normalized arc-length parametrization of a piecewise linear curve can be used.

Unit Cube. In the case of a map to the cube, $\psi : \mathcal{S} \rightarrow [0, 1]^3$, and a bijection is easily obtained by the 2D version of our method, or simply by means of Tutte's embedding.

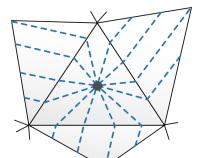
Unit Ball. When mapping M to the ball, $\psi : \mathcal{S} \rightarrow S^2$, the unit sphere. If \mathcal{S} is chosen to be the inner section, obtaining ψ is easy: \mathcal{S} is the boundary of a single tetrahedron, and one can map each triangular face to one half of a hemisphere. Alternatively, one could choose the outer boundary as \mathcal{S} , but then the numerically robust construction of a bijective ψ is more involved [Gotsman et al. 2003; Saba et al. 2005].

5.4 Minor Coordinate

As outlined in Section 3.1, each leaf ℓ is to be bijectively mapped to its corresponding leaf in a canonical foliation. This is done by defining a normalized parametrization $t_s : \ell_s \rightarrow [0, 1]$ for each leaf ℓ_s . Together, $\Psi(p) = (\psi(s_p), t_{s_p}(p))$ is a bijective map to the corresponding canonical leaf.

To this end, the lengths of the projection segments through the n -simplices need to be computed and summed. These lengths can be computed, in terms of Section 5.2, as $|\ell(\lambda_0) - \ell(\lambda_1)| = |(\lambda_0 - \lambda_1)d_i|$, i.e. they can be expressed as a fraction of the Euclidean length h of d_i , namely $|\lambda_0 - \lambda_1|h$. This leads to an arc-length parametrization of ℓ .

Note that in the polar cases the virtually removed center cell needs to be taken into account: the radius of the disk/ball that the center triangle/tetrahedron is mapped to (cf. Section 5.3) must be added to the leaf lengths so as to treat the leaves as extending to the cell's center point rather than ending/starting at its boundary. Any positive number can be chosen as radius; we set it to the cell's inradius. Furthermore, points p within the center cell need some special treatment: they need to be mapped to the boundary of the center cell first in order to determine the leaf they lie on.



Proposition 7. Let ψ be a continuous bijection. $t_s(p)$, for a leaf ℓ intersecting \mathcal{S} in a point s , is defined as a function of the arclength r_p between s_p and p , mapping each leaf to $[0, 1]$, i.e. we can write $t_{s_p}(r_p) = t_{s_p}(p)$. Assume that $t_s(r)$ is bijective and continuous both in r and s . Then $\Psi(p) = (\psi(s_p), t_{s_p}(r_p))$ is continuous (see Appendix A.5 for a proof).

5.5 Inverse

Evaluating the inverse $\Psi^{-1}(q)$ is easy as well. Using ψ^{-1} we locate the point in S that corresponds to the major coordinate(s) of q . Let ℓ be the leaf that starts at this point. To evaluate ϕ^{-1} , we trace the leaf ℓ , measuring its total length, and then trace it up to the relative length given by the minor coordinate of q . The point where this tracing ends is $p = \Psi^{-1}(q)$.

5.6 Numerics

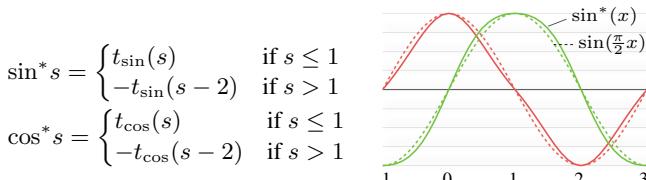
While in Section 3 we dealt with a purely combinatorial algorithm, the tracing of leaves and the determination of major and minor coordinates involves geometric computations, raising the question of numerical robustness.

General calculations using real numbers cannot be performed exactly or even just consistently in practice [Yap 1997]. In the field of rational numbers, however, exact calculation is possible. Several programming languages have built-in support for this, for others libraries are available that make working with rationals almost as easy as working with classical floating point numbers.

Note that, mainly due to our use of barycentric representations, all functions appearing in this section for leaf tracing and coordinate computation are rational. Hence, if their arguments p , d , and h are rational, so are their results. Thus, if the components of p and d are given in rational barycentric coordinates, and h is set to rational values, all computations can be carried out exactly using a rational number type. Note that a true arclength parametrization of the leaves requires $h = |d|$, which is generally irrational. However, we can set h to a rational approximation of $|d|$. In fact, any choice $h > 0$ leads to a bijective, though non-arclength, parametrization of the leaves. As h is constant per n -simplex and the fraction of it computed when tracing (cf. Section 5.4) varies continuously between leaves, continuity of the map is not affected either.

Rational Polar Coordinates. There is one situation though where irrational functions are involved: in the disk and ball map cases, one might want to convert the polar parametrization to a Cartesian one. Due to the involved trigonometric functions, this relates rational with irrational numbers. We can, however, use rational substitutes \sin^* , \cos^* , effectively defining an alternative relation between polar and Cartesian coordinates; the only requirement is that $(\sin^*\theta, \cos^*\theta)$, like $(\sin\theta, \cos\theta)$, is a homeomorphism between some interval $[a, b)$ and the cut unit circle. To achieve this, we define \sin^* and \cos^* as rational functions $[-1, 3] \rightarrow [-1, 1]$ as follows:

$$\text{Let } t_{\sin}(t) = \frac{2t}{1+t^2} \text{ and } t_{\cos}(t) = \frac{1-t^2}{1+t^2}.$$



6 Quality Considerations

The described method generates maps with the property of being bijective. Complementary to that is the search for maps of high quality in terms of distortion, e.g. with respect to isometry. In practice, the combination of both aspects is important.

Our algorithm for the construction of a discrete foliation described in Section 3 has a number of degrees of freedom: the source and sink region (red and green in Figures 3 and 4) can be chosen, the

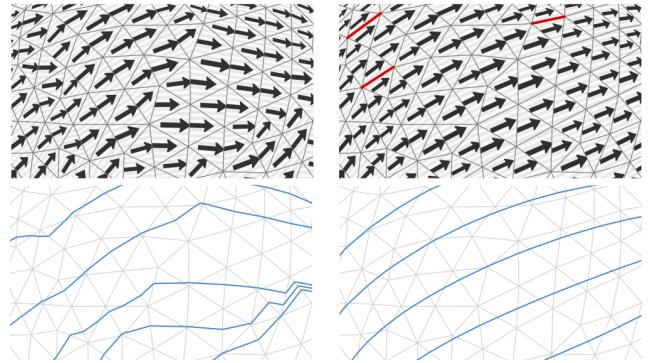


Figure 7: Constrained smoothing of an initial field (left) leads to a foliation with straighter leaves (blue) (right). The red edges are the ones whose duals' orientation has been reversed during smoothing.

shelling order can be varied (line 3 of Algorithm 1), and the directions d_i can be chosen subject to constraints (cf. Section 5.1). We can exploit these degrees of freedom to reduce distortion. While these options generally allow for a drastic improvement of the parametrization quality, there are obvious limits. Direct optimization with respect to a specific measure is difficult in the context of shellings and constrained direction fields. Map optimization methods that operate on a PL representation of the map are more flexible in this regard, cf. Section 7.

Source and Sink. In the polar cases, the sink (or source) is the entire boundary. We choose a well-centered inner source (or sink) by virtually removing the n -simplex with maximum distance to the boundary. In the case of a square map (cf. Figure 3), we divide the boundary into four parts of equal length. For a cube map, we need to choose two simply-connected regions on ∂M as source and sink. For our experiments, we used an automatic choice: we use iso-contours of the $[0,1]$ -normalized Fiedler vector (the globally smoothest non-constant scalar field, in least squares Laplacian sense). The iso- $\frac{1}{4}$ and iso- $\frac{3}{4}$ contours are natural choices for the source and sink region boundary. The Fiedler vector is the eigenvector corresponding to the second-smallest eigenvalue of the Laplace-Beltrami operator on ∂M (which can very efficiently be computed [Wu et al. 2014]), and the so obtained source and sink regions are generally simply-connected if a discrete Laplacian with positive edge weights is used [Fiedler 1975].

Shelling Order. In line 3 of Algorithm 1, a free n -simplex on the advancing front is to be chosen. As often most simplices are free, many choices are possible. In an attempt to keep the front smooth, we choose to guide it by a harmonic field. This field is computed as a minimizer of the Dirichlet energy subject to the boundary conditions that it vanishes at the source section and has value 1 at the sink section. Then, at each iteration, the advancing front algorithm always selects the free simplex with lowest field value.

Direction Field. Generally, we expect a direction field d with lower tangential curvature to lead to a map with lower distortion. Within the constraints implied by the shelling order, we seek the smoothest direction field. To this end, we initially set the directions to the centroid of their bounds (cf. Section 5.1). Then a few constrained Gauss-Seidel iterations of Laplacian smoothing are applied, maintaining correct signs of the dot products of the field with face normals by clamping. For additional improvement, this process is alternated with local shelling order changes: if in one iteration, the two directions d_i, d_j to both sides of an $(n-1)$ -simplex f_{ij} had to be clamped against f_{ij} , the shelling order is adjusted by flipping the orientation of the corresponding dual edge in G (if local and global bipolarity is preserved). This allows the field to become even smoother in the following iterations. Figure 7 illustrates the effect of field smoothing.

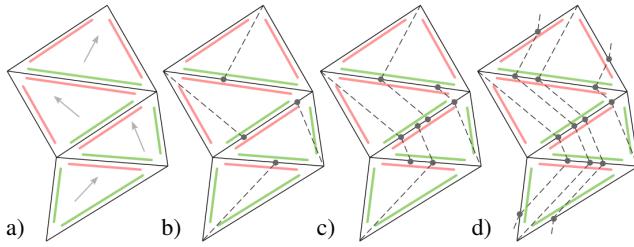


Figure 8: PL decomposition: a) original mesh. b) after initial refinement. c) after 1 level of refinement. d) final conforming result.

7 Piecewise Linear Representation

In Section 5, we defined a bijective map Ψ and saw how it can be evaluated for an arbitrary given point. For various purposes, a piecewise linear (PL) approximation Ψ_N of such maps, which is completely defined by the map values at the vertices of a simplicial mesh N (possibly different from M), is of interest. This representation should still be bijective and continuous, i.e. a *PL homeomorphism* is sought. Such a representation, for instance, enables application of various mesh-based map quality optimization methods which are able to preserve injectivity if the input is an injective map [Schüller et al. 2013; Hormann and Greiner 2000; Degener et al. 2003; Fu et al. 2015; Jin et al. 2015; Lipman 2012; Bommes et al. 2013].

Obtaining an approximation with this property by sampling a point-wise defined homeomorphism is a hard problem with no general solution in the multi-dimensional case [Groff 2003]. Our map Ψ defined in the previous section, however, has a very specific structure, which we can exploit to devise a simplicial mesh N such that the map values at its vertices define a PL homeomorphism,

$$\Psi_N(p = \alpha v_0 + \beta v_1 + \gamma v_2) := \alpha \Psi(v_0) + \beta \Psi(v_1) + \gamma \Psi(v_2),$$

where $\alpha + \beta + \gamma = 1$ are the barycentric coordinates of p in a triangle with vertices v_0, v_1, v_2 (analogously four components in the 3D case). Note that M itself is not in general a suitable mesh for this purpose; refinement can be inevitable. This is not a consequence of our specific foliation-based approach: it can be impossible to use certain input meshes for PL representation directly, as discussed in Section 2.

7.1 PL Structure Extraction

2D case. The refinement N of M is obtained using the following two-step procedure.

Initial Refinement: In each triangle c_i , add an additional edge parallel to d_i and incident to the one of the three vertices for which it actually lies in c_i . Afterwards, the mesh is non-conforming: vertices have been added on edges of triangles, and these vertices are, in general, not incorporated in the opposite triangles.

Iterative Refinement: As long as there is a triangle c_i with a non-incorporated vertex v on one of its edges: add an edge across c_i parallel to d_i , starting at v and ending at the opposite edge of c_i . Figure 8 illustrates the steps of this process. Note that the inserted edges form exactly those leaves which run through vertices of M . This implies that the process terminates.

The resulting mesh consists of triangles and quadrilateral faces. We split the quadrilateral faces into triangles by inserting diagonal edges. The resulting triangle mesh is called N and has the following important property: the two leaves through any two points within the same face of N run through the same sequence of triangles. These sequences of faces in N can be imagined as discrete, thick leaves (running between the dashed lines in Figure 8 right).

3D case. As in the 2D case, we can refine the input mesh M to a mesh N such that the leaves through any two points within the same cell of N run through the same sequence of tetrahedra. In the 2D case, this was achieved by splitting M by the integral curves which run through vertices. In 3D, we achieve this by splitting M by the integral surfaces which run through edges, i.e. the edges' extrusions parallel to d . This process splits each tetrahedron into a number of (truncated) polygonal prisms. We can then subdivide each such prism into tetrahedra: triangulate its base, inducing a subdivision of the polygonal prism into (truncated) triangular prisms, i.e. tetrahedra, pyramids, and prisms. Each pyramid can then be subdivided into two tetrahedra, and each prism into three tetrahedra.

Proposition 8. Ψ_N is a PL homeomorphism (cf. Appendix A.6).

This specific mesh N is not of immediate practical use, as it typically is very verbose (cf. Figure 9). As can be seen from the statistics in Figure 10, N can be orders of magnitude larger than M in the 2D case. In the 3D case, the situation is similar: for a mesh M with 5K tetrahedra, N typically has around 5M tetrahedra. Nevertheless, knowledge of the existence of N is of great practical interest. It enables us to devise algorithms that selectively refine M to M' only where the corresponding PL homeomorphism is undersampled, thus non-injective. The primary issue in this context is establishing termination of the refinement. By performing the selective refinement in a manner such that it progresses towards N , convergence to a PL homeomorphism is guaranteed trivially.

In Appendix B, we demonstrate a selective refinement strategy for the 2D case that follows exactly this principle and discuss additional options for obtaining practical alternatives to N .

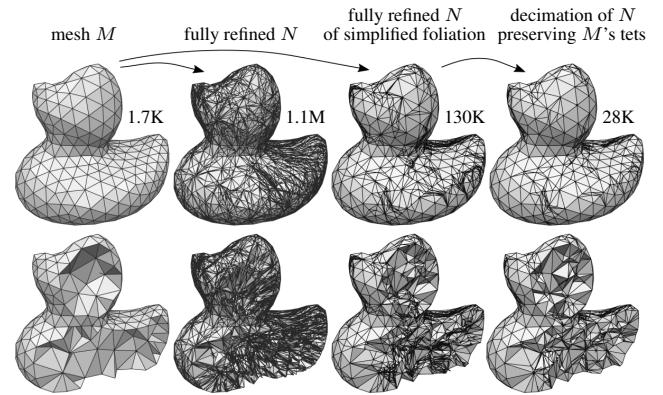


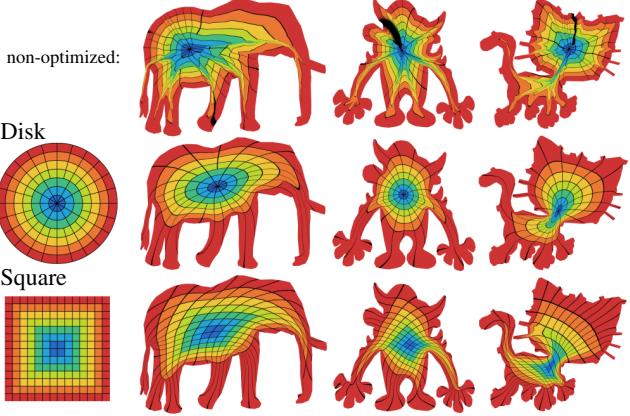
Figure 9: Tetrahedral mesh under various forms of PL refinement (cf. Section 7.1 and Appendix B). The bottom row shows cut views.

8 Results

The foliation, parametrization, and PL approximation algorithms were implemented in C++ using GMP [Granlund et al. 2015] for rational arithmetics.

Robustness. As we have mentioned in Section 4.4, we have tested the method on a large set of tetrahedral meshes, to validate the conjecture that a vast majority of meshes encountered in practice are shellable. At least for two types of objects (user-constructed content from Thingiverse and scanned content from AIM@SHAPE), meshed using standard methods, these assumptions appear to be confirmed.

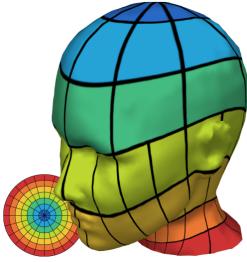
Quantitative comparisons to other methods are difficult as most use variations of harmonic maps which we found to produce non-bijective results in most cases, and code availability is limited. The most reliable techniques [Aigerman and Lipman 2013; Kovácsy et al. 2014] require specifying a distortion bound. We



	$ F $	type	full ref.	sel. ref.	t_{refine}	$t_{\text{opt}} \times 300$
Elephant	1796	disk	36440	567	1.2s	0.2s
		square	76898	664	0.5s	1.1s
Monster	5821	disk	204077	1987	6.8s	0.4s
		square	328304	1808	2.2s	2.5s
Cat	2169	disk	45590	597	1.8s	6.2s
		square	69077	523	0.5s	0.5s
Head	11342	disk	777748	148	3.3s	0.4s

Figure 10: Visualization of 2D bijections to the unit disk or unit square. They were turned into PL maps using selective refinement, cf. Section B.2, and optimized using a simple variant of [Hormann and Greiner 2000] using exact rational arithmetic. The table reports the number of input triangles ($|F|$), the number of triangle splits due to full refinement to N (full ref.) and selective refinement to M' (sel. ref.), the selective refinement runtime (t_{refine}), and the runtime of a constrained PL map optimization iteration (t_{opt}).

have tested 10 models with the code for [Aigerman and Lipman 2013] available online, with fixed bijective boundary maps to the sphere, and a range of values of the distortion bound $K \in \{20, 50, 100, 200, 500, 1000, 2000\}$. In Figure 12, we include three models (rightmost column) for which that method has failed to produce a bijection for every value of K . In Figures 10 and 12, we show visualizations of bijective maps generated using our implementation for a number of 2D and 3D shapes. Statistics are provided in Figure 10 and Table 1. Note that the method does not depend on M 's embedding. In particular, the 2D variant can be applied to 2-manifolds embedded in \mathbb{R}^3 , as depicted here.



In Figure 11 some examples of foliation leaves in complex shapes are shown.

Run Time. Our implementation of Algorithm 1 for finding a trivial foliation-compatible direction field processes approximately 30K tets/s. Constrained field smoothing processes \sim 25K tets/s in each iteration, and we generally find 10 iterations to be sufficient. The tracing of a leaf then proceeds through a sequence of around 250K tets/s. The map can thus typically be evaluated for a given point in under 1ms in meshes consisting of up to a million tetrahedra.

These timings have been measured on a 2014 Intel Core i7 Laptop. We expect that our prototype implementation can be accelerated significantly by several optimizations. For instance, we employed general data structures for arbitrary polytopal complexes of mixed dimension rather than a specialized simplicial mesh type. We also treated the various topological types of foliations all with the same generic code and used rational arithmetic without any kind of adaptive filtering or optimized predicates.

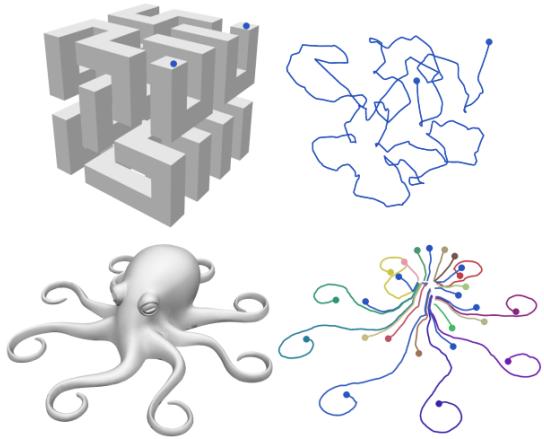


Figure 11: Visualization of some foliation leaves in a foliation with cube map (top) and ball map (bottom) topology.

Map Quality. The primary goal of our construction is to guarantee bijectivity for a maximally broad range of meshes, rather than to outperform existing approaches in terms of speed or map distortion. For instance, clearly, in case of success, a discrete harmonic mapping is hard to beat in terms of computation speed, and other methods, e.g. [Aigerman and Lipman 2013; Kovalsky et al. 2014] are hard to beat in terms of map distortion.

As described in Section 6, the map quality is taken into account indirectly (smoothness of the shelling front, smoothness of the field). The resulting map is thus less smooth than those typically obtained by other, e.g. harmonic map, approaches. This can be seen in the wrinkles and bumps in the iso-surfaces. A way to address this is via a piecewise linear representation of the map, which can be used as a starting point for a variety of algorithms, constrained to maintain bijectivity, as demonstrated for the 2D case in Figure 10.

9 Limitations and Future Work

By design, the method handles only simply-connected domains. We note that only such domains are shellable, and the combinatorial theory will have to be changed in order to extend it to higher genus without having to partition the domain into simple parts.

While there is a guarantee that for any non-shellable tetrahedral mesh shellable refinements exist, there is no known practical algorithm that is guaranteed to find one efficiently in general.

The quality of the map is constrained by the combinatorics of the shelling, and may be far from optimal especially on very coarse meshes of complex shape, e.g. in Figure 11 top.

	$ T $	t_{shell}	t_{field}	$10 \times$	t_{trace}
Bunny	17K	0.5s	0.7s	0.42ms	
Hippo	12K	0.4s	0.6s	0.26ms	
Hand	10K	0.3s	0.5s	0.36ms	
Crystal	97K	3.6s	4.3s	0.37ms	
Dumbbell	11K	0.3s	0.5s	0.25ms	
Cat	35K	1.2s	1.4s	0.42ms	
Fandisk	42K	1.6s	2.0s	0.38ms	
Fandisk	103K	3.1s	4.7s	0.85ms	
Fandisk	724K	31.2s	32.4s	0.91ms	
Lever (Fig. 1)	92K	2.5s	4.0s	0.77ms	

Table 1: Statistics for the 3D volumetric models in Figure 12: the number of tetrahedra, the time t_{shell} taken by Algorithm 1, the time t_{field} taken by an iteration of constrained direction field smoothing, and the time t_{trace} needed to evaluate the map Ψ for a given point (or any set of multiple points with constant major coordinates).

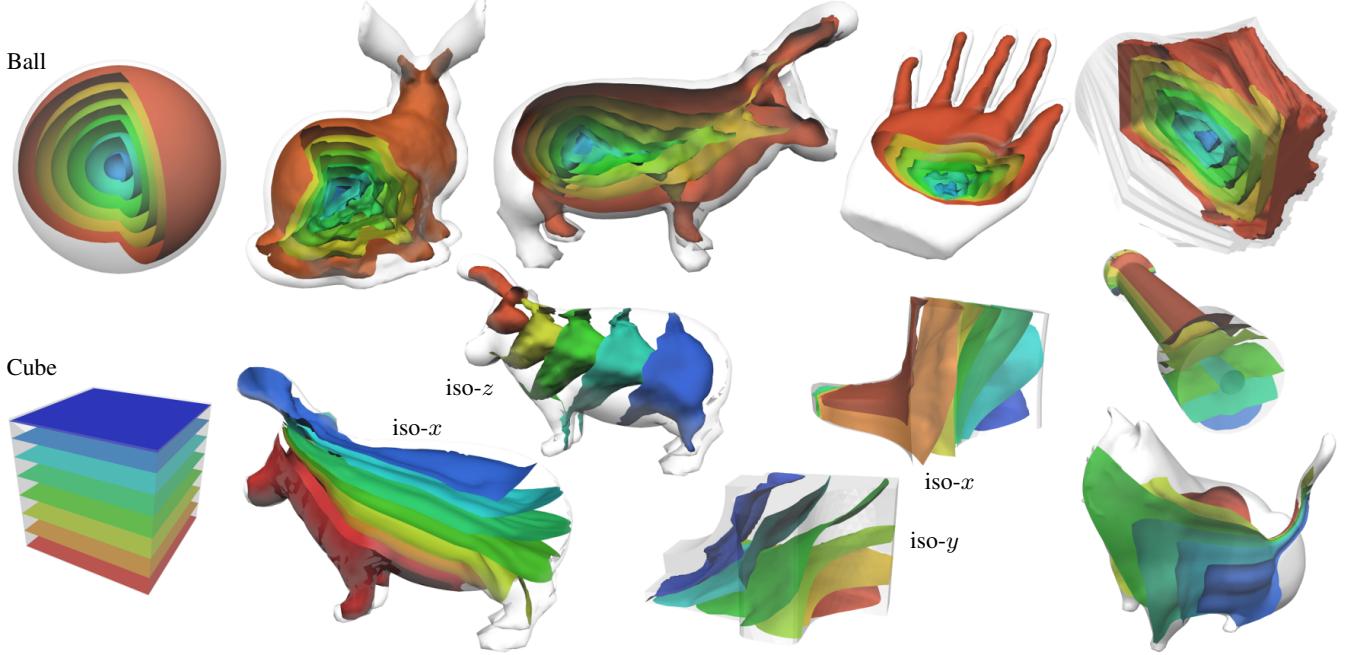


Figure 12: Isosurface visualization of bijective parametrizations over the unit ball or unit cube obtained with the method introduced in Sections 3, 5, 6 (Ψ is shown, PL approximation or mesh-based optimization have not been used).

Future Work. The major remaining challenge on the theory side is to identify a way to ensure that a mesh has a shelling order, either by integrating this into the tessellation process, by (adaptive) refinement, or by partitioning the mesh into simpler parts and mapping the parts individually.

Using different foliation topologies it should be possible to handle maps to more general domains like the sphere, torus, or annulus. Extension to higher genus, perhaps via branched Riemannian balls [Zeng et al. 2007], is certainly of interest as well.

Of particular practical value is the investigation of efficient selective refinement algorithms for the 3D case, akin to the one presented for the 2D case, that generate parsimonious PL approximations of the bijection Ψ . The question of minimal refinement is certainly of interest as well. Furthermore, after establishing bijections Ψ_1 and Ψ_2 from M_1 and M_2 to a common domain, a bijection from M_1 to M_2 is immediately available as $\Psi_2^{-1}\Psi_1$. A PL representation thereof can be obtained by means of intersection of PL representations of Ψ_1 and Ψ_2 [Alexa et al. 2000; Weber and Zorin 2014], but an approach that performs refinement in an integrated manner rather than this two-stage process could be more efficient.

We focused on showing how a bijection can be constructed. For various applications, more control is required. In particular, constraints that fix the value of certain points are commonly of interest. For points on the boundary of the domain, such constraints can be handled via a prescribed source section boundary map. The development of a general solution also for internal points, perhaps based on a domain decomposition strategy, is clearly an interesting path for future work.

To the best of our knowledge, no method has been described in the literature that is able to construct 3D vector or direction fields strictly adhering to a given topology (in particular: a prescribed set of singularities). Close to this goal is, e.g., the work by Theisel et al. [2004], which can enforce critical points but not prevent additional ones. Our method for constructing trivial foliation-compatible direction fields achieves this for certain special cases. It is worthwhile exploring whether this can be extended to a more general setting.

Acknowledgments

This work was supported in part by NSF awards IIS-1320635, CNS-1229185, and CCF-1533564, the Moore-Sloan Data Science Environment at NYU, AT&T, DARPA, MLB Advanced Media, and an IBM Faculty Award. The authors would like to thank Jonas Costa for help with comparisons, Mirela Ben-Chen and Ofir Weber for providing data sets.

References

- ADIPRASITO, K. A., AND BENEDETTI, B. 2012. Subdivisions, shellability, and collapsibility of products. *arXiv:1202.6606*.
- ADIPRASITO, K. A., AND IZMESTIEV, I. 2015. Derived subdivisions make every PL sphere polytopal. *Israel Journal of Mathematics* 208, 1, 443–450.
- AIGERMAN, N., AND LIPMAN, Y. 2013. Injective and bounded distortion mappings in 3D. *ACM Trans. Graph.* 32, 4, 106:1–106:14.
- AIGERMAN, N., AND LIPMAN, Y. 2015. Orbifold tutte embeddings. *ACM Trans. Graph.* 34, 6, 190:1–190:12.
- AIGERMAN, N., PORANNE, R., AND LIPMAN, Y. 2015. Seamless surface mappings. *ACM Trans. Graph.* 34, 4, 72:1–72:13.
- ALEXA, M., COHEN-OR, D., AND LEVIN, D. 2000. As-rigid-as-possible shape interpolation. In *Proc. SIGGRAPH ’00*, 157–164.
- ALEXANDER, J. W. 1924. On the subdivision of 3-space by a polyhedron. *Proc. Nat. Acad. Sci. of the USA* 10, 1, 6.
- BOMMES, D., CAMPEN, M., EBKE, H.-C., ALLIEZ, P., AND KOBBELT, L. 2013. Integer-grid maps for reliable quad meshing. In *Proc. SIGGRAPH 2013*, 98:1–98:12.
- CHOQUET, G. 1945. Sur un type de transformation analytique généralisant la représentation conforme et défini au moyen de fonctions harmoniques. *Bulletin des Sciences Mathématiques* 69, 156–165.

- COLIN DE VERDIÈRE, E., POCCHIOLA, M., AND VEGTER, G. 2003. Tutte's barycenter method applied to isotopies. *Computational Geometry* 26, 1, 81 – 97.
- DE FRAYSSEIX, H., DE MENDEZ, P. O., AND ROSENSTIEHL, P. 1995. Bipolar orientations revisited. *Discrete Applied Mathematics* 56, 2–3, 157 – 179.
- DEGENER, P., MESETH, J., AND KLEIN, R. 2003. An adaptable surface parameterization method. *Int. Meshing Roundtable* 3, 201–213.
- DEY, T. K., EDELSBRUNNER, H., AND GUHA, S. 1999. Computational topology. In *Advances in Discrete and Computational Geometry*, 109–143.
- FIEDLER, M. 1975. A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czechoslovak Mathematical Journal* 25, 4, 619–633.
- FILIPPOV, A. F. 1988. *Differential equations with discontinuous righthand sides*. Mathematics and its Applications. Kluwer Academic Publ, Dordrecht.
- FLOATER, M. S., AND PHAM-TRONG, V. 2006. Convex combination maps over triangulations, tilings, and tetrahedral meshes. *Advances in Computational Mathematics* 25, 4, 347–356.
- FLOATER, M. S. 1997. Parametrization and smooth approximation of surface triangulations. *Computer Aided Geometric Design* 14, 3, 231 – 250.
- FLOATER, M. S. 2003. One-to-one piecewise linear mappings over triangulations. *Math. Comput.* 72, 242, 685–696.
- FU, X.-M., LIU, Y., AND GUO, B. 2015. Computing locally injective mappings by advanced MIPS. *ACM Trans. Graph.* 34, 4, 71:1–71:12.
- FURCH, R. 1924. Zur Grundlegung der kombinatorischen Topologie. In *Abhandlungen aus dem mathematischen Seminar der Universität Hamburg*, vol. 3, Springer, 69–88.
- GERGEN, J. J. 1930. Mapping of a general type of three dimensional region on a sphere. *American Journal of Mathematics* 52, 2, 197–224.
- GERGEN, J. J. 1931. Note on the green function of a star-shaped three dimensional region. *American Journal of Mathematics* 53, 4, 746–752.
- GOODRICK, R. E. 1968. Non-simplicially collapsible triangulations of I^n . In *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 64, Cambridge Univ Press, 31–36.
- GOTSMAN, C., GU, X., AND SHEFFER, A. 2003. Fundamentals of spherical parameterization for 3D meshes. In *Proc. SIGGRAPH 2003*, 358–363.
- GRANLUND, T., ET AL. 2015. *GNU MP: The GNU Multiple Precision Arithmetic Library*, 6.1.0 ed. <http://gmplib.org/>.
- GROFF, R. E. 2003. *Piecewise linear homeomorphisms for approximation of invertible maps*. PhD thesis, The University of Michigan.
- HORMANN, K., AND GREINER, G. 2000. MIPS: An efficient global parametrization method. In *Curve and Surface Design 1999*, Innovations in Applied Mathematics. 153–162.
- HYUNH, L., AND GINGOLD, Y. I. 2015. Bijective deformations in \mathbb{R}^n via integral curve coordinates. *CoRR abs/1505.00073*.
- JIN, Y., HUANG, J., AND TONG, R. 2014. Remeshing-assisted optimization for locally injective mappings. *Computer Graphics Forum* 33, 5, 269–279.
- JIN, Y., QIAN, G.-P., ZHAO, J.-Y., CHANG, J., TONG, R.-F., AND ZHANG, J. 2015. Stretch-minimizing volumetric parameterization. *J. Comp. Sci. and Tech.* 30, 3, 553–564.
- KING, S. 2004. How to make a triangulation of S^3 polytopal. *Trans. Am. Math. Soc.* 356, 11, 4519–4542.
- KNESER, H. 1926. Lösung der Aufgabe 41. *Jahresbericht der Deutschen Mathematiker-Vereinigung* 35, 123–124.
- KOVALSKY, S. Z., AIGERMAN, N., BASRI, R., AND LIPMAN, Y. 2014. Controlling singular values with semidefinite programming. *ACM Transactions on Graphics* 33, 4.
- LAUGESEN, R. S. 1996. Injectivity can fail for higher-dimensional harmonic extensions. *Complex Variables Theory Appl.* 28, 4, 357–369.
- LEMPEL, A., EVEN, S., AND CEDERBAUM, I. 1967. An algorithm for planarity testing of graphs. In *Theory of Graphs*. Gordon and Breach, 215–232.
- LI, X., GUO, X., WANG, H., HE, Y., GU, X., AND QIN, H. 2007. Harmonic volumetric mapping for solid modeling applications. In *Proc. SPM '07*, 109–120.
- LIN, J., XIA, J., GAO, X., LIAO, M., HE, Y., AND GU, X. 2015. Interior structure transfer via harmonic 1-forms. *Multimedia Tools and Applications* 74, 1, 139–158.
- LIPMAN, Y. 2012. Bounded distortion mapping spaces for triangular meshes. *ACM Trans. Graph.* 31, 4, 108:1–108:13.
- LIU, H., AND LIAO, G. 1996. A note on harmonic maps. *Applied Mathematics Letters* 9, 4, 95 – 97.
- MANI, P., AND BRUGGESSER, H. 1971. Shellable decompositions of cells and spheres. *Mathematica Scandinavica* 29, 197–205.
- MOERDIJK, I., AND MRČUN, J. 2003. *Introduction to Foliations and Lie Groupoids*. Cambridge Studies in Advanced Mathematics. Cambridge University Press.
- MOISE, E. E. 1952. Affine structures in 3-manifolds: V. The triangulation theorem and Hauptvermutung. *Annals of Mathematics*, 96–114.
- MYLES, A., PIETRONI, N., AND ZORIN, D. 2014. Robust field-aligned global parametrization. *ACM Trans. Graph.* 33, 4.
- NGUYEN, T., AND JÜTTLER, B. 2010. Parameterization of contractible domains using sequences of harmonic maps. In *Curves and Surfaces*, vol. 6920 of *LNCS*, 501–514.
- PACHNER, U. 1991. PL homeomorphic manifolds are equivalent by elementary shellings. *Europ. J. Comb.* 12, 2, 129–145.
- RAY, N., AND SOKOLOV, D. 2014. Robust polylines tracing for n-symmetry direction field on triangulated surfaces. *ACM Trans. Graph.* 33, 3.
- RUDIN, M. E. 1958. An unshellable triangulation of a tetrahedron. *Bulletin of the American Mathematical Society* 64, 3, 90–91.
- SABA, S., YAVNEH, I., GOTSMAN, C., AND SHEFFER, A. 2005. Practical spherical embedding of manifold triangle meshes. *Int. Conf. on Shape Modeling and Applications*, 258–267.
- SCHNEIDER, T., AND HORMANN, K. 2015. Smooth bijective maps between arbitrary planar polygons. *Computer Aided Geometric Design* 35, 243–254.
- SCHÜLLER, C., KAVAN, L., PANZZO, D., AND SORKINE-HORNUNG, O. 2013. Locally injective mappings. In *Proc. SGP '13*, 125–135.

- SMITH, J., AND SCHAEFER, S. 2015. Bijective parameterization with free boundaries. *ACM Trans. Graph.* 34, 4, 70:1–70:9.
- STODDART, A., ET AL. 1964. The shape of level surfaces of harmonic functions in three dimensions. *The Michigan Mathematical Journal* 11, 3, 225–229.
- TUTTE, W. T. 1963. How to draw a graph. *Proc. Lond. Math. Soc.* 13, 743–767.
- WANG, Y., GU, X., AND YAU, S.-T. 2004. Volumetric harmonic map. *Communications in Information & Systems* 3, 3, 191–202.
- WEBER, O., AND ZORIN, D. 2014. Locally injective parametrization with arbitrary fixed boundaries. *ACM Trans. Graph.* 33, 4, 75:1–75:12.
- WEINKAUF, T., THEISEL, H., HEGE, H.-C., AND SEIDEL, H.-P. 2004. Topological construction and visualization of higher order 3D vector fields. *Computer Graphics Forum* 23, 3, 469–478.
- WU, J., SONG, J., AND ZHANG, W. 2014. An efficient and accurate method to compute the Fiedler vector based on Householder deflation and inverse power iteration. *J. Computational Applied Mathematics* 269, 101–108.
- XIA, J., HE, Y., HAN, S., FU, C.-W., LUO, F., AND GU, X. 2010. Parameterization of star-shaped volumes using Green’s functions. In *Advances in Geometric Modeling and Processing*, vol. 6130 of *LNCS*. 219–235.
- XIA, J., HE, Y., YIN, X., HAN, S., AND GU, X. 2010. Direct-product volumetric parameterization of handlebodies via harmonic fields. In *Proc. SMI 2010*, 3–12.
- YAP, C. K. 1997. Robust geometric computation. In *Handbook of Discrete and Computational Geometry, chapter 35*. CRC Press LLC, Boca Raton, FL, 653–668.
- ZENG, W., LI, X., YAU, S.-T., AND GU, X. 2007. Conformal spherical parametrization for high genus surfaces. *Communications in Information & Systems* 7, 3, 273–286.
- ZHANG, E., MISCHAIKOW, K., AND TURK, G. 2006. Vector field design on surfaces. *ACM Trans. Graph.* 25, 4, 1294–1326.
- ZIEGLER, G. M. 1998. Shelling polyhedral 3-balls and 4-polytopes. *Discrete & Computational Geometry* 19, 2, 159–174.

A Appendix

A.1 Proof of Proposition 2

Proof. We proceed by showing the following equivalence:

c is a local source (sink) in $G_f \Leftrightarrow c$ is incoming (outgoing) for f .

First, observe that an n -simplex c is incoming for an incident k -simplex f , $k < n - 1$, if and only if it is incoming for all simplices of dimension $k + 1$, incident at c and f . By induction, it is incoming for all simplices of dimension m , $k < m < n$, shared by c and f .

Consider G_f , the dual graph G restricted to the star of a k -simplex f . If an n -simplex is a source relative to G_f , it is incoming for all its $(n - 1)$ -simplices which are also incident at f , and inductively for f itself. We conclude that a local source simplex in G_f is an incoming simplex for f . Analogously, any local sink simplex is an outgoing simplex for f . Conversely, if an n -simplex is incoming for f , it is incoming for the $(n - 1)$ -simplices shared by it and f . It is thus a local source in G_f . Analogously, any outgoing simplex for f is a local sink in G_f . Therefore, considering Corollary 1, the conditions for foliation compatibility and local bipolarity are equivalent. \square

A.2 Proof of Proposition 4

Proof. For every k -simplex f , $k < n$, there is a local source in G_f , because the n -simplex $c_i \in G_f$ with the lowest i is a local source. Assume there are multiple sources in G_f . Let c_i and c_j , $j > i$, be the first two. Consider the prefix $P_j = c_1, \dots, c_j$. In the star of f in P_j the simplex c_j forms a separate connected component (if there was an adjacent n -simplex c_l , c_j would not be a source due to $l < j$ and $c_l \rightarrow c_j$). As c_i also is in the star of f in P_j , f is singular in P_j , thus P_j not manifold, contradicting that in a bishelling P_j actually is manifold. By symmetry of bishellings, the same arguments apply for local sinks. We conclude that G is locally bipolar.

As the implied graph orientation follows the linear bishelling order, G is acyclic. The first simplex c_1 clearly is a global source (there is no predecessor in the order), c_m is a global sink. Assume there is another global source $c_i \in G$, $i > 1$. Consider the prefix $P_i = c_1, \dots, c_i$. Simplex c_i is isolated in P_i (if it had a face-adjacent n -simplex c_l , $l < i$, it would not be a global source). Thus P_i has multiple components. As P_m has only one component, there must be a P_j , $j \geq i$, where these multiple components merge, necessarily first coming into contact in a singular vertex or edge, i.e. P_j is not manifold, contradicting that in a bishelling P_j actually is manifold. By symmetry of bishellings, the same arguments apply for global sinks. We conclude that G is also globally bipolar. \square

A.3 Proof of Proposition 5

Proof. If c_1, \dots, c_m is a shelling, c_0, \dots, c_m clearly is a shelling. We need to show that it is a reverse shelling as well, i.e. that the prefix $U_i = c_0, \dots, c_i$ is manifold for each $i < m$. Assume the opposite, i.e. there is an i such that U_i is not manifold. Then there is a singular vertex or edge on the boundary of U_i . As this boundary is internal to the extended complex, the suffix $V_{i+1} = c_{i+1}, \dots, c_m$ (the complement of U_i) has the same boundary with the same singular vertices or edges (because their immediate neighborhoods are complementary to those in U_i). This contradicts V_i being manifold for each i , which follows from c_0, \dots, c_m being a shelling. \square

A.4 Proof of Proposition 6

Proof. We consider the 3D case. Due to Proposition 5 we can focus on the cube map case. We outline a proof that proceeds almost precisely along the lines of the proof in [Moise 1952] for shellability. As M is PL-homeomorphic to a 3-ball, there is a PL map on a subdivision of M which maps it to a cube piecewise linearly with source and sink areas mapping to two opposite faces F and F' of the cube: any 3D finite simply-connected simplicial complex M is PL-homeomorphic to a tetrahedron, and therefore to a cube, and any map of the cubes’ surface to itself can be extended to a PL-homeomorphism of the cube to itself. We can easily define a subdivision of a cube (and its image under a PL map of the cube to the tetrahedron), that has a bishelling order with sources in F and sinks in F' : e.g., for a cube add a vertex in the center, and split the cube into six pyramids, each pyramid into two tetrahedra. The required bishelling order starts with two tetrahedra at F , then adds tetrahedra in pyramids with bases in cube faces sharing an edge with F , followed by tetrahedra in the pyramid with base in F' .

From this elementary bishelling, we can obtain a bishelling of any k -th barycentric subdivision of the cube, in particular, of the one that coincides combinatorially with a subdivision of M .

As a barycentric subdivision can be performed by stellar refinement of one k -simplex at a time, $k \leq 3$, we only need to show that the original order can be extended to the order on a mesh with a single simplex stellated. In the refined mesh, if two tetrahedra c_1 and c_2 belong to different tetrahedra of the original mesh, then they are

ordered in the same sequence as parent tetrahedra. It remains to order them in a single refined tetrahedron.

The bishelling order on the original mesh defines sink and source region on the surface of each tetrahedron, each forming a disk K and K' respectively, consisting of one to three faces. As a result of a single stellar refinement, the tetrahedron is split into 4, 3 or 2 tetrahedra S_i . Our goal is to define a sequence of S_i , such that marking sink and source faces for tetrahedra following this order results, at every step, in satisfying the condition that $S_i \cap \cup_{j < i} S_j \cup K$ is a topological disk, and $S_i \cap \cup_{j > i} S_j \cup K'$ is a topological disk. Using the idea from [Moise 1952], we can reduce the number of cases one needs to consider using collapses.

We also only need to consider the cases of number of faces in the source and sink disk equal to $(1, 1)$, $(1, 2)$, $(1, 3)$, $(2, 1)$, $(2, 2)$, as all other cases are obtained by symmetry. Observe that for a 1-4 split, one can always take as first tetrahedron in the bishelling order a tetrahedron with a base on a face f of K ; the remaining 3 tetrahedra form a mesh that can be deformed by a PL homeomorphism into a 1-3 split configuration, moving a vertex in the center of the tetrahedron to the center of f . Thus the problem of finding a shelling sequence for a 1-4 refined tetrahedron is reduced to 1-3 refinement in all cases. Unless there is only one source and one sink face, further reduction is possible, by a collapse to a 1-2 split configuration. Thus, we need to consider 1-2 splits for four different source-sink face number pairs, $(1, 2)$, $(1, 3)$, $(2, 1)$, $(2, 2)$, which may have different locations with respect to the split edge, and 1-3 splits with respect to $(1, 1)$. This forms 13 cases for which the existence of ordering of the two or three tetrahedra S_i satisfying the required condition can be checked directly.

Thus, the elementary bishelling order can be extended to a bishelling order on the refined mesh at a single stellar refinement step, and continued in this way to an r -refinement of the tetrahedron, isomorphic to a refinement of M . After adding virtual cells c_{source} and c_{sink} , covering the refined source and sink regions of the refined M , this order extends to a $c_{\text{source}}c_{\text{sink}}$ -bishelling. \square

A.5 Proof of Proposition 7

Proof. A different way to state that a PC direction field is foliation-compatible is that the corresponding differential equation has unique solutions for all initial points in the domain M . As a consequence, foliation-compatible PC direction fields satisfy the conditions of [Filippov 1988, Corollary 1, Ch. 2, §8], which states that in this case the solutions of the ODE $\dot{p} = d(p)$, $p(0) = p_0$ (understood as continuous solutions satisfying the equation at all points where d is defined), continuously depend on the initial parameter, i.e., for any $\epsilon > 0$, there is δ , such that if $|p_0 - p'| < \delta$, then $|p(r) - p'(r)| < \epsilon$ everywhere in the domain; in other words, in this case the standard continuous dependence on parameters applies, too (note the additional requirement of solution uniqueness, which does not hold for all PC fields, but for foliation-compatible PC fields).

As our fields have unit length, the solution of the ODE is arclength parameterized. Consider a neighborhood of a point p of M consisting of points q of M such that $s(q) \in U_\epsilon(s(p))$, the neighborhood of radius ϵ of $s(p)$ in \mathcal{S} , and with $r(p) - \epsilon < r(q) < r(p) + \epsilon$. We pick it to be small enough, so that for any r , $|r - r(p)| < \epsilon$, and fiber passing through $U_\epsilon(p)$, it has a point with arclength $r(p)$.

These observations ensure that the map $G : B_\epsilon(s(p)) \times [r(p) - \epsilon, r(p) + \epsilon] \rightarrow M$ defined by $(p_0, r) \rightarrow p(r)$, is continuous; composing it on the left with $H : (s, r) \rightarrow (s, t^{-1}(r))$, which is also continuous (as t is a continuous map between compact sets), yields a continuous map on each set $B_\epsilon(s(p)) \times [t^{-1}(r(p) - \epsilon), t^{-1}(r(p) + \epsilon)]$. This map is globally defined and continuous on $\mathcal{S} \times [0, 1]$, as for each leaf, $t(p)$ maps the leaf to $[0, 1]$. Composing this map with $(s, r) \rightarrow (\psi^{-1}(s), t)$, which is also continuous, we finally

obtain the inverse of Ψ , which also has to be continuous by compactness. \square

A.6 Proof of Proposition 8

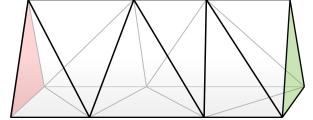
Proof. We consider the 3D case. N has a specific structure, which can be thought of as discrete foliation, with thick leaves formed by chains of tetrahedra (analogous to the 2D situation with chains of triangles, depicted in Figure 8 right). We call these tetrahedra chains *stacks*.

To see this, first notice that, by construction, each tetrahedron has exactly one edge which is aligned with a leaf of the field. It cannot have multiple, because the field is constant in each tetrahedron of M , thus in each of N , and it cannot have none because it would have been split by one of its edges' integral surface.

Consider a triangle $f_0 = (v_1, v_2, v_3)$ in \mathcal{S}^r , the refined source region, and the three leaves running from $v_i \in \mathcal{S}^r$ to the sink section. We show that there is a sequence of tetrahedra in N , c_i with $i = 0 \dots k$, with all vertices on the three leaves, that are pairwise face-adjacent, i.e. there is a sequence of faces f_i , $i = 0, \dots k+1$, starting with f_0 , such that f_{i+1} is shared by c_i and c_{i+1} . This sequence forms a stack.

There is a single tetrahedron c_0 which has f_0 as a face. As c_0 has one edge on a leaf, its vertex v_4 not contained in the face f_0 is on one of the three leaves through the vertices of f_0 . W.l.o.g. assume that it is the leaf passing through vertex v_1 . Define f_1 to be the face of c_0 with vertices v_2, v_3, v_4 . Remove c_0 from the mesh. Now we have exactly the same configuration we had with f_0 with respect to f_1 , and can construct f_2 in the same way. As we always proceed forward along the three leaves, this process terminates when we reach a sink face. As stacks starting from adjacent faces of \mathcal{S}^r share their leaves, these stacks necessarily form a partition of N , i.e. each tetrahedron of N belongs to one stack.

Now consider the case of a cube map, with $D = [0, 1]^3$. The images of leaves are parallel straight line segments. The figure on the right shows a stack of tetrahedra in parameter space, i.e. part of $\Psi(N)$, spanned by three parallel straight edge chains (leaves) running from source (red) to sink (green). We need to show that all tetrahedra of this $\Psi(N)$ -stack are oriented consistently.



In a stack, let v_1^i, v_2^i, v_3^i be the 3 vertices of f_i that are on the three leaves ℓ_1, ℓ_2, ℓ_3 of the stack. This ordering of vertices defines an orientation on the planes of all f_i 's, i.e. a choice of normals $(v_2 - v_1) \times (v_3 - v_1)$. We choose the numbering so that the positive side of f_i is the one towards the interior of c_i . Then, because all tetrahedra have positive volume in N by construction, the fourth vertex of c_i is always on the positive side of f_i . In other words, $\det(v_2 - v_1, v_3 - v_1, v_4 - v_1) > 0$ for all tetrahedra in the stack.

Let $w_i = \Psi(v_i)$, for $i = 1, 2, 3$, denote the vertex images, and s_i the point where the leaf through v_i intersects the source \mathcal{S} . We define $w_i^0 = \Psi(s_i)$. Because the leaf images are straight, we can write $w_i = w_i^0 + t_i e$, where $t_i \in \mathbb{R}$, and $e \in \mathbb{R}^3$ is the constant leaf direction vector in parameter space. For the fourth vertex, which shares its leaf with the first, we have $w_4 = w_1^0 + t_4 e$. By direct computation we observe that $\det(w_2 - w_1, w_3 - w_1, w_4 - w_1) = (t_4 - t_1) \det(w_2^0 - w_1^0, w_3^0 - w_1^0, e)$. Hence, as $t_4 > t_1$, a tetrahedron in $\Psi(N)$ inherits its orientation from the orientation of its stack's base triangle in $\Psi(\mathcal{S})$.

This immediately implies that all tetrahedra in the same stack have the same orientation. Note that $\Psi(\mathcal{S}) = (\psi(\mathcal{S}), 0)$. Hence, assuming that ψ is a PL homeomorphism on the source section, i.e., that it

maps the triangulation \mathcal{S}^r bijectively (which, e.g., for a Tutte map ψ always is the case), also across stacks tetrahedra have the same orientation. The fact that all tetrahedra in $\Psi(N)$ have the same orientation together with a non-selfintersecting domain boundary implies bijectivity of Ψ [Aigerman and Lipman 2013]. \square

To extend the analysis to the ball case, one needs to consider non-parallel leaves. We assume that the directions of the three leaves of a stack are all contained in a hemisphere. In this case, the stack images are triangular pyramid frustums, which can be mapped to prisms by linear transformations, such that the same arguments can be used to show that the map is a bijection. If the leaves passing through an individual face of \mathcal{S}^r span more than a hemisphere, an additional split of the stack on top of it is necessary.

B Approximation of PL Maps

The mesh N resulting from the full refinement procedure can be very large. For reasons of parsimony and efficiency, it is interesting to consider ways of creating a coarser PL homeomorphism $\Psi_{M'}$ approximating Ψ . There are several options to this end, as detailed in the following.

B.1 Homeomorphism Decimation

We can use operators for triangle and tetrahedral mesh decimation, such as edge-collapses, to decimate N to a simpler mesh M' . In a typical mesh decimation approach, one collapses an edge only if doing so does not result in inverted elements. As we are dealing with not just a mesh but a PL homeomorphism, we must check that both, the elements of N and their images under Ψ , remain positively oriented. In other words, we are effectively dealing with two isomorphic meshes, N and $\Psi(N)$, that are decimated in sync while avoiding inversions. Figure 9 shows an example, where the decimation of N was restricted such that the mesh remains a refinement of M , i.e. the underlying structure of M is preserved.

B.2 Selective Refinement

A disadvantage of the decimation approach is the high complexity of the intermediate mesh N . As a more practical alternative, we can approach the problem from the opposite direction: instead of decimating the fully refined mesh N , refine the original mesh M only as far as necessary. We thus consider Ψ_M , a potentially *undersampled*, non-bijective map, and refine M where it is undersampled until it becomes bijective, thus a homeomorphism $\Psi_{M'}$. Here, the key difficulty is ensuring convergence of the refinement process. Exploiting our knowledge that N is a suitable mesh for a homeomorphism, however, we can design a selective refinement algorithm that progresses towards N , thereby guaranteeing termination. We demonstrate this for the 2D case in the following.

We call the triangles of M master-triangles and the parts of them resulting from refinement sub-triangles. A triangle whose image is inverted or degenerate is called violating. The incremental refinement procedure is fairly simple: as long as there is a violating (master- or sub-)triangle, project the vertex for which the triangle is incoming or outgoing to the opposite edge, insert a new vertex there, and re-triangulate the two adjacent master-triangles into sub-triangles to incorporate this new vertex. An illustration is shown in Figure 13.

To see the correctness of this algorithm, first note that a triangle c_i (or one of its sub-triangles) with d_i parallel to one of its edges is never violating. This follows from the two incident vertices having the same major coordinate in this case, as detailed in Appendix A.6. The insertion of the parallel-projected vertex causes the two resulting sub-triangles of c_i to have an edge parallel to d_i . For this, we only need to make sure that vertices on opposite edges of a

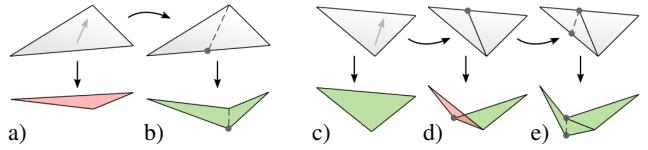


Figure 13: Selective refinement. a) if a triangle is violating, i.e. its image is degenerate or inverted (red), a vertex is projected and inserted. b) afterwards, the two sub-triangles are non-violating (green). c) if a non-violating triangle is refined (d) because the refinement of a neighboring triangle caused a new vertex, its sub-triangles can become violating. e) projecting a vertex and inserting it again remedies the violation.

triangle which are parallel projections of each other are connected by an edge in the sub-triangulation of the triangle.

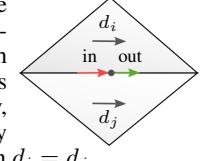
Proposition 9. The 2D selective refinement algorithm terminates.

Proof. In each step, a new vertex is inserted that is one of the vertices of the fully refined mesh N . As N is finite, only finitely many steps can be performed. \square

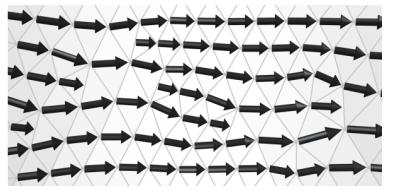
B.3 Foliation Simplification

Both approaches described, decimation and selective refinement, in some way rely on the PL structure given by N . This raises the question of whether we can exploit the degrees of freedom in the construction of the foliation to make N simpler in the first place.

N is obtained from M by splitting along every vertex's integral curve (every edge's integral surface). If we could design the foliation such that multiple vertices (edges) share, i.e. lie on, the same integral curve (surface), this would simplify N by reducing the overall number of splitting curves (surfaces). We can do this in the following way: if we set the field d_i, d_j parallel to an edge f_{ij} between triangles c_i and c_j , the two vertices incident to f_{ij} lie on a common integral curve. Likewise, if we set d_i, d_j parallel to a 2-simplex f_{ij} between tetrahedra c_i and c_j , the three edges incident to f_{ij} lie on a common integral surface. Note that in these cases, deviating from Section 4, both n -simplices are incoming (and outgoing) for points on the common $(n-1)$ -simplex f_{ij} . However, they both provide the same integral curve, i.e. this does not violate foliation-compatibility. Effectively, the two n -simplices can be treated as virtually merged because they have a common direction $d_i = d_j$.



In terms of the oriented dual edge graph G , this corresponds to collapsing the dual edge $c_i \rightarrow c_j$, virtually merging the two n -simplices. Of course we cannot do this everywhere, the oriented graph needs to remain locally and globally bipolar. We thus propose a greedy dual edge removal strategy. The oriented dual edge graph is taken as input, and dual edges are collapsed one by one, ordered by the deviation their virtual collapse induces on the field (due to adjusting d to be parallel), unless their collapse violates bipolarity (which is easily tested locally). The inset shows an example result field; note that most direction vectors are now aligned with mesh edges. The effect of foliation simplification on N is demonstrated in Figure 9.



Note that the rational approximation of $h \approx |d|$ (cf. Section 5.6) needs to be consistent (i.e. equal) in n -simplices adjacent to an $(n-1)$ -simplex to which the field is parallel in order to preserve continuity of the final map.