DOI: 10.1111/cgf.12864
EUROGRAPHICS 2016
J. Madeira and G. Patow
(Guest Editors)

Volume 35 (2016), Number 2
STAR – State of The Art Report

Directional Field Synthesis, Design, and Processing

Amir Vaxman[1] Marcel Campen[2] Olga Diamanti[3] Daniele Panozzo[2,3] David Bommes[4] Klaus Hildebrandt[5] Mirela Ben-Chen[6]

[1]Utrecht University [2]New York University [3]ETH Zurich [4]RWTH Aachen University [5]Delft University of Technology [6]Technion

Abstract

Direction fields and vector fields play an increasingly important role in computer graphics and geometry processing. The synthesis of directional fields on surfaces, or other spatial domains, is a fundamental step in numerous applications, such as mesh generation, deformation, texture mapping, and many more. The wide range of applications resulted in definitions for many types of directional fields: from vector and tensor fields, over line and cross fields, to frame and vector-set fields. Depending on the application at hand, researchers have used various notions of objectives and constraints to synthesize such fields. These notions are defined in terms of fairness, feature alignment, symmetry, or field topology, to mention just a few. To facilitate these objectives, various representations, discretizations, and optimization strategies have been developed. These choices come with varying strengths and weaknesses. This report provides a systematic overview of directional field synthesis for graphics applications, the challenges it poses, and the methods developed in recent years to address these challenges.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—

1. Introduction

An increasing number of computer graphics and geometry processing methods rely on, or are guided by, spatially-varying directional information, assigned to each point on a given domain. These *directional fields* exist in many flavors: some specify a magnitude in addition to a direction, while others consider multiple directions per point, often with some notion of symmetry among them. Directional fields appear in the literature under several names, such as *vector fields*, *direction fields*, *line fields*, *cross fields*, *frame fields*, *RoSy fields*, *N-symmetry fields*, *PolyVector fields*, or *tensor fields*. We provide a taxonomy of the different variants, discuss, and compare their properties. We use the term "directional field" to refer to the general class of such fields, and use more specific definitions in the context of the respective literature.

A directional field can be the result of a (real-world or virtual) measurement of the geometric or physical properties of an object, or its surface. Notable examples are the principal directions of a shape, stress or strain tensors, the gradient of a scalar field, the advection field of a flow, and diffusion data from MRI. There exists a large body of literature exploring ways to *analyze* (and visualize) such fields, including comprehensive surveys [LHZP07, BCP*12]. We are instead interested in surveying the body of work that focuses on the active creation and processing of such fields, in the context of geometry processing and computer graphics. Directional fields can be *synthesized* (also: *designed*) by a computational model that considers user constraints, alignment conditions, fairness objectives, or physical realizations.

There have been significant developments in directional field synthesis over the past decade. These developments have been driven by the increasing demand for applications that require directional fields, in their diverse variants. Prominent examples include: surface parametrization, mesh generation, texture synthesis, flow simulation, fabrication, architectural geometry, and illustration.

Different applications have different requirements. To name a few examples, some applications require the prescription of a specific field topology, whereas other applications infer it automatically; some require that the field is integrable to a scalar function, whereas others require that the flow of the vector does not generate distortion;

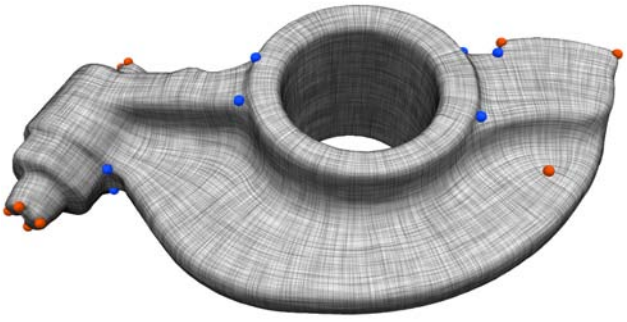


Figure 1: Visualization of a directional field that was synthesized based on fairness and alignment objectives. Its singularities are depicted by little dots, colored according to their index.

some require a soft alignment with curvature directions, whereas others require hard alignment with certain user constraints.

While there exists a plethora of algorithms for synthesizing directional fields, there is no "one-size-fits-all" method which is applicable in all cases. With a given set of *objectives* and *constraints*, the main design choices are for the most appropriate *representation* and *discretization* scheme for directional quantities, and for an *optimization* strategy to achieve the design goals. The intricate interplay between these various choices makes it challenging to find the best approach, given specific application requirements. The goal of this report is to clarify the implications of these choices, guide practitioners to the right choice, and encourage researchers to address the (multitude of) remaining open questions.

A recent course on vector field processing [dGDT15] provides an additional introduction to the topic, with a focus on the aspects of discretization (vertex vs. edge vs. face based) of vector fields.

1.1. Reading Guide

This report covers a wide range of material, and treats a large number of independent aspects. It has been structured to enable a linear reading order, which we recommend for readers interested in a comprehensive in-depth understanding of the topic. A selective, non-linear reading is possible as well, as the text is well-equipped with forward and backward pointers between sections. We offer the following suggestions for readers that are interested in specific areas.

Computer graphics and shape modeling Section 9 serves as a dictionary to specific applications. Especially relevant for computer graphics is the usage of directional fields in deformation (Section 9.2) and texture mapping (Section 9.3). For shape modeling, relevant applications are mesh generation (Section 9.1), deformation (Section 9.2) and architectural geometry (Section 9.4). Reading Section 2 allows to understand the typical types of directional fields in use. To go more in depth, read Section 8 for a description of "fair" directional field design. To learn more about the nature and limitations of discrete fields on meshes, read Section 6.

Geometry processing and discrete differential geometry Proficient researchers can typically read this report in linear progression. Nevertheless, we refer those who are specifically interested in discrete vector calculus to Section 7, and those who are interested in topological analysis to section 6. Sections 4 and 5 provide a necessary background to the "who" and the "where" of directional fields, whereas Section 8, and subsequent Section 11, provide a more practical account of the "how".

Physical simulation You should find an interest in discrete vector calculus and discrete differential operators, described in Section 7, where Sections 4 and 5 provide the necessary background. Important representations of directional fields for simulations are tensors (Section 5.3), 1-forms (Section 5.5) and linear operators (Section 5.7). Mesh generation (Section 9.1) is a particularly relevant application that might be of interest. Take note of Section 8.3 for details on setting differential constraints, such as the divergence or the curl of a field, and of Section 10 for the visualization of the results.

Signal processing To get acquainted with directional field representation and sampling, read Section 3 for the continuous notions, and then follow through the sections up to and including Section 7. Of particular interest are Sections 4.4 and 6 that explain the effects of a discrete representation on the topology of the field.

Big data and visualization Section 10 details the existing methods to visualize directional fields on surfaces. The comparative analysis in 11.4, and the paragraph about scalability within, discuss aspects of directional field synthesis on increasingly complex meshes.

1.2. Overview

In addition to providing a comprehensive overview of the recent contributions that have been made to this topic, we establish a structured categorization of directional fields. Furthermore, we analyze important and interesting aspects that have not been explicitly discussed before. In particular, we cover the following topics:

Types of Directional Fields We classify the distinct types of directional fields used in the literature in Section 2. They differ by a number of parameters, such as the number of directional entities per point of the domain, symmetries between them, and whether they encode magnitude in addition to direction. A precise notation is introduced, in order to avoid confusion between many terms that are used ambiguously.

Differential Geometry The mathematical formalism of directional fields in the continuous setting provides the theoretical foundation for computational synthesis in the discrete setting. This is covered in Section 3.

Discretization One can think of "discretization" as *where* directional fields are represented. For instance, the directional information can "live" on the supporting planes of the faces of a triangulation, on discrete tangent planes defined on vertices, or as scalar integrated 1-forms on edges. A choice of discretization can retain some properties of directional fields from the continuous setting, such as their differential or their topological structure, but usually not all of them. Furthermore, discrete representations can be viewed as a *sampling* of a continuous field, and are thus liable to effects such as aliasing. We treat these aspects in Section 4.

Representation We define "representation" as *how* directional fields are encoded. In $\mathbb{R}^2$, an explicit representation using Euclidean coordinates is straightforward. However, the situation is more complicated on curved surfaces. To handle this, a large variety of representations for directional fields has been explored. This variety ranges from representations based on local Cartesian or polar coordinates, through discrete 1-forms and complex number-based representations, to more indirect encoding, e.g. as the roots of polynomials, or the maxima of scalar functions. These are described in Section 5.

Topology and Operators Given where and how directional fields are encoded, we proceed to describe how their topological and differential properties are formulated in the discrete setting. We discuss the discrete definitions of directional-field singularities in Section 6. We show how operators from vector calculus can be in defined in the discrete setting in Section 7.

Objectives and Constraints We describe common measures of quality for directional fields, and means to prescribe required properties. A popular measure of quality is fairness, though other types of objectives and constraints also appear in the literature. These include: alignment with a sparse or dense set of directional constraints, symmetry, or adherence to a specific topology. We present the various types of synthesis objectives, and discuss the amenability of the different representations to these goals in Section 8.

Applications While our main focus is the general problem of directional field synthesis, we outline specific application scenarios in Section 9. The wide range of applications reveals the variety of different requirements posed on directional field synthesis, which led to the multitude of diverse treatments of directional data.

Field Visualization A visual understanding of the synthesized fields is often helpful, or even a necessity. Various effective visualization techniques have been developed for directional fields. We briefly present them in Section 10.

Algorithms and Comparison We provide a desiderata-based guide to choosing the right method for various purposes, and empirically compare some of the properties of the state-of-the-art methods, in Section 11.

Open Questions We conclude in Section 12 with an outlook on future research, by presenting open problems, shortcomings, and remaining questions.

2. Types of Directional Fields

Directional fields come in many different flavors. Unfortunately, the available terminology in the literature suffers from many inconsistencies—some terms are synonymous, some are homonymous, and others are simply ambiguous and context-dependent. In light of this, we introduce a notation that allows us to unambiguously refer to specific types of fields. For the purpose of familiarity, we indicate common names used in the literature for these fields.

We refer to a directional object (in short: a directional) as a "direction" if the magnitude is irrelevant, and as a "vector" if it plays a role. A *field* on a *domain* is the assignment of a directional to each point in the domain.

A directional field can be multi-valued, describing a set of directions or vectors at every point. Our only assumption is that the size of the set, denoted as N, is constant throughout the field. We assume this since a setting with varying N has found no application so far. The cases of $N = 1,2,4,6$ are the most common in practice. Of particular interest are *rotationally-symmetric* direction fields, or in short: *RoSy fields*. Common variants are two directions with π-radians RoSy, four directions with $\pi/2$ RoSy, or two independent pairs of directions with π RoSy within each pair. These symmetry properties are very natural in many applications, for instance when dealing with principal curvature directions [HZ00], principal directions of stress or strain tensors [PTP*15], conjugate directions [LXW*11, DVPSH14], or Langer's lines [MPP*13, BLP*13], to name a few.

We encode the type of the field using a set of integers $\{r_1,\ldots,r_k\} \in \mathbb{N}^k$, whose sum is the size of the N-set ($\sum_i r_i = N$). These r_i indicate that the N-set is partitioned into k subsets with cardinalities r_i, and within each subset the directions or vectors obey rotational symmetry, i.e. there are angles $2\pi/r_i$ between them. Furthermore, in the case of vectors, the elements of each such subset are equal in magnitude. We contract multiple equal values for brevity: if $r_i = r_{i+1} = \cdots = r_{i+m-1}$, we write $(r_i)^m$.

Common examples include:

1-vector field	One vector, classical "vector field"
2-direction field	Two directions with π symmetry, "line field", "2-RoSy field"
1^3-vector field	Three independent vectors, "3-polyvector field"
4-vector field	Four vectors with $\pi/2$ symmetry, "non-unit cross field"
4-direction field	Four directions with $\pi/2$ symmetry, "unit cross field", "4-RoSy field"
2^2-vector field	Two pairs of vectors with π symmetry each, "frame field"
2^2-direction field	Two pairs of directions with π symmetry each, "non-ortho. cross field"
6-direction field	Six directions with $\pi/3$ symmetry, "6-RoSy"
2^3-vector field	Three pairs of vectors with π symmetry each

3. Differential Geometry of Directional Fields

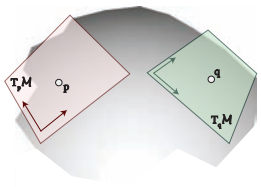
In order to perform a systematic study of directional fields in discrete settings, we give a concise introduction to the theory of continuous directional fields on manifolds, covering definitions of basic concepts. It is considerably out of the scope of this report to include a full description. Therefore, every section includes references to textbooks for a comprehensive account.

Note that most theoretical concepts only pertain to 1-vector fields. The relevant properties of general directional fields are covered in more detail in the subsequent sections.

3.1. Differential and Riemannian Structure

In this section, we review basic notions concerning the geometry of surfaces. For a comprehensive introduction to Riemannian geometry, we refer the reader to [dC92, Kue05, Jos08].

Tangent Bundle and Vector Fields We consider a smooth, compact and oriented 2-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^3$. For any point $p \in \mathcal{M}$, the tangent space $T_p\mathcal{M}$ of $\mathcal{M}$ at p is a two-dimensional vector space. Any tangent vector at p is orthogonal to the surface normal of $\mathcal{M}$ at p. Hence, we can identify $T_p\mathcal{M}$ with



the subspace of $\mathbb{R}^3$ that is orthogonal to the surface normal of $\mathcal{M}$ at p (see inset). The union

$$T\mathcal{M} = \bigcup_{p\in\mathcal{M}} T_p\mathcal{M}$$

of all tangent spaces forms a 4-dimensional manifold, called the *tangent bundle* of $\mathcal{M}$. Locally it is trivial, which means that around every point $p \in \mathcal{M}$ there is an open neighborhood $U \subset \mathcal{M}$ such that $\bigcup_{p\in U} T_p\mathcal{M}$ is diffeomorphic to $U \times \mathbb{R}^2$. Every vector $v \in T\mathcal{M}$ lies in one of the tangent spaces $T_p\mathcal{M}$, and we call the corresponding point p the foot point of v. The projection $\pi : T\mathcal{M} \mapsto \mathcal{M}$ maps every vector in the tangent bundle to its foot point. A *tangent vector field* on $\mathcal{M}$ is a *section* of the tangent bundle: a smooth map $\mathbf{v} : \mathcal{M} \mapsto T\mathcal{M}$ such that $\pi \circ \mathbf{v} : \mathcal{M} \mapsto \mathcal{M}$ is the identity. For further reading, we refer to [AF02, Chapter 3.2] and [Jos08, Chapter 2].

Cotangent Bundle and 1-forms The dual space of a vector space consists of the linear maps from that space to $\mathbb{R}$. The dual space is again a vector space of the same dimension as the primal space. We denote the dual spaces of the tangent spaces by $T_p\mathcal{M}^*$. The union of all *cotangent spaces,* $T\mathcal{M}^* = \bigcup_{p\in\mathcal{M}} T_p\mathcal{M}^*$, forms the *cotangent bundle*. A section in the cotangent bundle is called a 1-*form*. For example, we can apply a 1-form ω to a vector field $\mathbf{v}$. The result $\omega(\mathbf{v})$ is a function on $\mathcal{M}$.

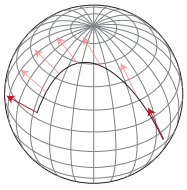
Connections and Parallel Transport An affine connection (or covariant derivative) associates with two tangential vector fields $\mathbf{v}$ and $\mathbf{w}$ a new tangential vector field $\nabla_{\mathbf{w}}\mathbf{v}$. This map is linear in $\mathbf{w}$

$$\nabla_{f\mathbf{w}_1+\mathbf{w}_2}\mathbf{v} = f\nabla_{\mathbf{w}_1}\mathbf{v} + \nabla_{\mathbf{w}_2}\mathbf{v}$$

and a derivation in $\mathbf{v}$

$$\nabla_{\mathbf{w}}(f\mathbf{v}_1 + \mathbf{v}_2) = (\nabla_{\mathbf{w}} f)\mathbf{v}_1 + f\nabla_{\mathbf{w}}\mathbf{v}_1 + \nabla_{\mathbf{w}}\mathbf{v}_2,$$

where $\mathbf{v}, \mathbf{v}_1, \mathbf{v}_2, \mathbf{w}, \mathbf{w}_1$ and $\mathbf{w}_2$ are smooth vector fields, f is a smooth function and $\nabla_{\mathbf{w}} f$ is the derivative of f in direction $\mathbf{w}$. We can think of $\nabla_{\mathbf{w}}\mathbf{v}$ as the derivative of $\mathbf{v}$ in direction $\mathbf{w}$.



Using an affine connection, we can define the parallel transport of a vector along a curve on the manifold. Consider a curve $c : [0,1] \mapsto \mathcal{M}$ and a vector $\mathbf{v}_0 \in T_{c(0)}M$. Then, there is a unique vector field $\mathbf{v} : [0,1] \to TM$ along c (which means $\pi(\mathbf{v}(t)) = c(t)$ for all t) that solves the linear differential equation $\nabla_{\dot{c}(t)}\mathbf{v}(t) = 0$ with the initial condition $\mathbf{v}(0) = \mathbf{v}_0$. The vector field $\mathbf{v}(t)$ is called the *parallel transport* of $\mathbf{v}_0$ along c. The inset figure shows an example of a vector that is parallel transported along a curve on the unit sphere. The parallel transport of vectors is linear: This means if a vector in $T_{c(t_0)}M$ is a linear combination of other vectors in $T_{c(t_0)}M$, it will be the same linear combination (same weighted sum) after the parallel transport of the vectors. For proofs, we refer to [dC92, Section 2.2].

Riemannian Metric A scalar product on a vector space provides a measure of vector norm (or length) and the angles between vectors. A Riemannian metric g on $\mathcal{M}$ assigns a scalar product $\langle\cdot,\cdot\rangle_p$ to any tangent space $T_p\mathcal{M}$. This assignment is smooth, i.e., the map $p \to \langle\cdot,\cdot\rangle_p$ is smooth. A Riemannian metric allows for defining various geometric concepts on a differentiable manifold, such as the distance of points in the manifold, geodesic curves, angles between pairs of vectors, intersection angles between curves, the volume of domains in the manifold, intrinsic curvatures (like the Gaussian curvature) and differential operators (including gradient, divergence, curl, Laplace operators). A manifold that is equipped with a Riemannian metric is called a Riemannian manifold.

We can construct a Riemannian metric on surfaces in $\mathbb{R}^3$ using the scalar product of $\mathbb{R}^3$. Every tangent plane of the surface is a subspace of $\mathbb{R}^3$. Hence, we obtain a scalar product on every tangent space by restricting the scalar product of $\mathbb{R}^3$ to the tangent plane. Note that since the surface normal is changing along the surface, the resulting Riemannian metric on the surface is not flat.

Levi-Civita Connection On a Riemannian manifold, we are interested in affine connections that satisfy

$$\nabla_{\mathbf{u}} g(\mathbf{v},\mathbf{w}) = g(\nabla_{\mathbf{u}}\mathbf{v},\mathbf{w}) + g(\mathbf{v},\nabla_{\mathbf{u}}\mathbf{w}), \tag{1}$$

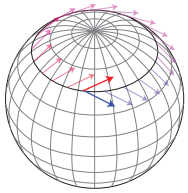
which means that the affine connection is compatible with the Riemannian metric g. For the parallel transport of vectors, this means that the scalar product between any pair of vectors does not change when the vectors are parallel-transported along a curve. Hence, for an affine connection that is compatible with g, the maps between two tangent spaces, obtained by parallel transporting vectors along any curve, are isometries. Then, the parallel transport of a vector along a curve can be described by the oriented angle that the vector forms with the tangent of the curve. The derivative of the oriented angle along the curve is the same for every vector that is parallel transported along the curve.

Among the affine connections that are compatible with a Riemannian metric g, there is a unique one that is torsion-free, *i.e.*, satisfies

$$\nabla_{\mathbf{w}}\mathbf{v} - \nabla_{\mathbf{v}}\mathbf{w} + [\mathbf{v},\mathbf{w}] = 0.$$

Here $[\mathbf{v},\mathbf{w}]$ denotes the Lie bracket of $\mathbf{v}$ and $\mathbf{w}$. This connection is called the *Levi-Civita connection* (or Riemannian connection). For the Levi-Civita connection, the parallel transport of vectors is linked to the geodesic curvature of the curve in the surface. The derivative of the oriented angle between the transported vector and the tangent of the curve equals the geodesic curvature of the curve. For proofs, we refer to [dC92, Section 2.3] and [O'N66, Section VII.3].

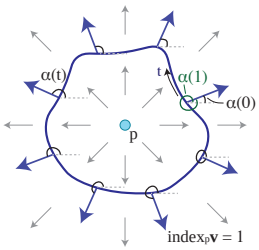
Holonomy Consider a closed curve c on a surface $\mathcal{M}$. Parallel transporting a vector $\mathbf{v}_p$ along c from a point p to itself does not in general yield the same vector $\mathbf{v}_p$ again. In the inset figure, the red vector is parallel transported along a circle of latitude on the sphere, yielding the blue vector. In this case, the vector rotates with constant speed relative to the tangent of the curve (in other words: the derivative of the angle between the vector and the tangent of the curve is constant). The oriented angular difference between $\mathbf{v}_p$ and its transport $\mathbf{v}'_p$ is called the *holonomy angle* of the curve, and is independent of p and $\mathbf{v}_p$. The holonomy angle of the Levi-Civita connection is closely related to the Gaussian curvature of the surface: the holonomy angle of a smooth curve that bounds a simply-connected domain equals the integral of the Gaussian curvature over the domain that is enclosed by the curve, modulo 2π. For proofs, we refer to [O'N66, Section 7.3].



3.2. Vector Field Topology

In this section, we consider topological properties of vector fields: singularities, indices, and the Poincaré–Hopf theorem. Proofs of the concepts and theorems presented in this section can be found in chapters 7 and 8 of [Ful95], which provides a good introduction to algebraic topology (including vector field topology).

A vector field has a singularity at a point p if it vanishes or is not defined at this point. We assume that the field has a finite number of singularities. Let us first consider a singular point p of a vector field $\mathbf{v}$ on a domain in $\mathbb{R}^2$, as shown in the inset figure.



We consider a small, simple (not self-intersecting), closed curve around p, parametrized (in counterclockwise direction) by a function $c : [0,1] \mapsto \mathbb{R}^2$. By "small", we mean that the (topological) disc enclosed by the circle does not contain a second singularity in the field. We inspect the vector field along the curve. Since none of the vectors $\mathbf{v}(c(t))$ vanishes, we can represent the vector field along the curve in polar form. This means there is a smooth *angle function* $\alpha : [0,1] \mapsto \mathbb{R}$, going counterclockwise around the curve, such that

$$\mathbf{v}(c(t)) = \|\mathbf{v}(c(t))\| \begin{pmatrix} \cos(\alpha(t)) \\ \sin(\alpha(t)) \end{pmatrix}.$$

The function α is not unique, since we can add multiples of 2π to α and get the same vectors $\mathbf{v}(c(t))$. However, since α is smooth, the difference $\alpha(1) - \alpha(0)$ is unique, it is a multiple of 2π, and it depends neither on the curve $c(t)$ (as long as it is simply-connected and does not contain a second singularity), nor on the starting point $c(0)$. We define the *index* of the singularity of $\mathbf{v}$ at p to be the integer

$$\text{index}_p \mathbf{v} = \frac{1}{2\pi}(\alpha(1) - \alpha(0)).$$

The index measures the number of times the vectors along the curve c rotate counterclockwise, while traversing the curve once. In the context of direction fields, it is common to consider only points p with $\text{index}_p \mathbf{v} \neq 0$ as singular. We adopt this herein.

The definition does not directly extend to surfaces, because there is no global coordinate system (the tangent bundle is not trivial). However, we can calculate the index at a singular point p of a vector field $\mathbf{v}$ on a surface $\mathcal{M}$ by using an arbitrary chart around p. The chart maps the vector field on a local neighborhood of p on the surface to a vector field on an open set of the plane. Following that, we can use the definition discussed above to compute the index, and this computation would be invariant to the specific choice of the chart.

A vector field cannot have an arbitrary set of singularities. For a surface without boundary, the sum of all indices is related to the genus g of the surface. Explicitly, the Poincaré–Hopf theorem states that the sum of all the indices of a vector field equals 2-2g.

The concept of indices of singularities can be generalized to other types of direction fields (Figure 2). In these cases, the index is not an integer anymore. For example, for N-vector fields, the index is a multiple of $\frac{1}{N}$ [RVLL08]. Similar to 1-vector fields, direction fields

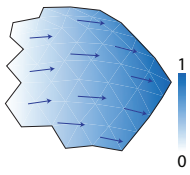
Figure 2: Singularities of index $\frac{1}{2}$, $\frac{1}{4}$, 0 (non-singular), $-\frac{1}{4}$, $-\frac{1}{2}$ in a 4-vector field. The black curves are the so-called separatrices – integral curves (cf. Section 10.3) of the field intersecting the singularity.

obey the Poincaré–Hopf theorem: they cannot have an arbitrary set of singularities, as the sum is the topological constant 2-2g. This has been described in [RVLL08] for N-direction fields, and in [DVPSH14] for 1^N-direction fields.

3.3. Vector Calculus

Vector calculus is concerned with the differentiation and integration of vector fields. This includes differential operators like gradient, divergence, curl and the Laplace operator. Most papers dealing with the processing of vector fields are, at least implicitly, using vector calculus. Our focus is on vector calculus on surfaces. It is closely related to exterior calculus, and all presented concepts could alternatively be formulated in terms of differential forms and operators on the exterior algebra. For brevity, we restrict the presentation to vector calculus. For an in-depth treatment of vector analysis and exterior calculus, and proofs of the concepts presented in this section, we refer to [War83, AF02]. A recommended undergraduate text is [Jae13].

Gradient The differential of a smooth function f is a 1-form.



In many cases, it is more convenient to work with a vector field describing the derivative instead. This vector field is called the *gradient* of f. We can think of the gradient of a function as the vector field that points to the direction of the steepest ascent of the function, as shown in the inset figure. Formally, the gradient of f is defined as the unique tangential vector field that satisfies

$$\langle \text{grad}\, f, \mathbf{v} \rangle = \nabla_{\mathbf{v}} f$$

for all tangential vector fields $\mathbf{v}$. We emphasize that for the construction of the gradient of a function a metric is needed.

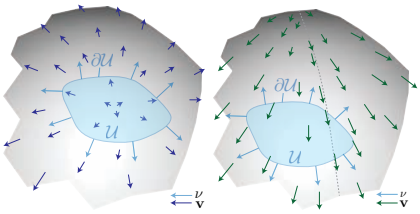
Divergence The *divergence* is a linear operator mapping vector fields to functions. At any point $p \in \mathcal{M}$, the divergence of a smooth vector field $\mathbf{v}$ is defined by

$$\text{div}\, \mathbf{v}(p) = \sum_{i=1}^{d} \langle \nabla_{\mathbf{e}_i} \mathbf{v}(p), \mathbf{e}_i \rangle \tag{2}$$

where $\{\mathbf{e}_i\}$ is an arbitrary orthonormal basis of $T_p\mathcal{M}$. Let $\mathcal{U}$ be a compact subset of $\mathcal{M}$ and ν be the outer-pointing normal at the boundary $\partial\mathcal{U}$, then

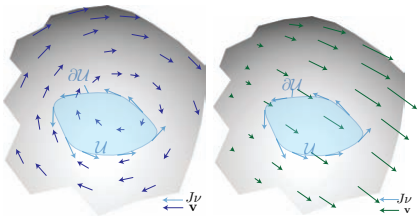
$$\int_{\mathcal{U}} f\, \text{div}\, \mathbf{v}\, dA = -\int_{\mathcal{U}} \langle \text{grad}\, f, \mathbf{v} \rangle\, dA + \int_{\partial\mathcal{U}} f \langle \nu, \mathbf{v} \rangle\, ds. \tag{3}$$

If we think of the vector field as a velocity field (*e.g.* of a fluid),



then the divergence of the vector field provides information about the sources and sinks of the flow. If we set f to be the constant unit function ($f(p) = 1\ \forall p$) in (3), the first term vanishes, and the equation shows that the integral of the divergence of a vector field measures the flow into and out of $\mathcal{U}$. Here, $\mathcal{U}$ can be any domain in the manifold. If the boundary integral is positive, there is more flowing out of than into the domain, which means that the domain is a source. Similarly, if the boundary integral is negative, the domain is a sink. The inset figure shows two examples of vector fields with non-vanishing divergence. In both cases the shown domain $\mathcal{U}$ is a source.

Curl For surfaces, the *curl* is closely related to the divergence. It measures the amount by which the field locally circulates around each point. Intuitively, it measures the amount by which a wheel placed at each point of the domain would spin, if forces were applied to it according to the vector field at that point. In the inset figure, both fields have non-vanishing curl.



To reveal the connection to the divergence, we consider the operator J that rotates any vector of a vector field in its tangent plane by $\frac{\pi}{2}$ (following the orientation of the surface). For a surface embedded in $\mathbb{R}^3$, we can represent this operator using the cross product and the surface normal field: $\mathrm{J}\colon \mathbf{v} \mapsto N \times \mathbf{v}$. The curl operator maps vector fields to functions and is defined by

$$\operatorname{curl} \mathbf{v} = -\operatorname{div} \mathrm{J}\, \mathbf{v}. \tag{4}$$

This means it measures the divergence of the field after a rotation of $\frac{\pi}{2}$ of all vectors in their respective tangent planes. Analogous to (3), the curl satisfies the equation

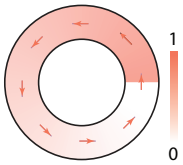
$$\int_{\mathcal{U}} f \operatorname{curl} \mathbf{v}\, \mathrm{d}A = \int_{\mathcal{U}} \langle \operatorname{grad} f, \mathrm{J}\, \mathbf{v} \rangle\, \mathrm{d}A + \int_{\partial \mathcal{U}} f \langle \mathrm{J}\, \mathbf{v}, \mathbf{v} \rangle\, \mathrm{d}s. \tag{5}$$

In the same manner as the divergence, by setting $f = 1$ in (5), we can see that the curl of a vector field measures how much the vector field circulates around the domain. Analogous to sources and sinks, the curl in the field is generated by vortices. The divergence measures the flow *in and out* of the domain, while the curl measures the flow *along* the boundary. The inset figure shows two vector fields with non-vanishing curl. In both cases, the flow circulates around the domain $\mathcal{U}$.

Hodge Decomposition and Harmonic Fields The space $\mathcal{X}$ of square-integrable tangential vector fields on a surface with vanishing boundary can be decomposed into three orthogonal subspaces

$$\mathcal{X} = \operatorname{Image}(\operatorname{grad}) \oplus \operatorname{Image}(\mathrm{J} \operatorname{grad}) \oplus \mathcal{H},$$

where the domain of the gradient is the Sobolev space $W^{1,2}$ of functions whose differential is square-integrable. The gradient fields have the property that their curl vanishes, and the rotated gradient fields have vanishing divergence. The remaining space $\mathcal{H}$ consists of the *harmonic vector fields*. These fields are both divergence and curl-free. Consequently, they are gradients of scalar functions in simply-connected subdomains (locally), but not otherwise (globally). An example is shown in the inset figure. For surfaces without boundary, the space of harmonic vector fields $\mathcal{H}$ equals the first singular cohomology of the surface. This is an important relation between vector calculus and algebraic topology. The dimension of the space of harmonic tangential vector fields on a surface of genus g and without boundary is $2g$. For a comprehensive treatment of the Hodge decomposition and proofs of the statements made above, we refer to [War83, Chapter 6].



For manifolds with boundary, analogous decompositions of spaces of vector fields for different types of boundary conditions are possible. For an in-depth treatment of the topic, we refer the reader to [Sch95].

Exterior Calculus Vector calculus is closely related to exterior calculus, and all presented concepts could alternatively be formulated in terms of differential forms and operators on the exterior algebra. We briefly discuss this relation. We denote the space of smooth differential i-forms on $\mathcal{M}$ by Λ^i, the exterior derivative by $\mathrm{d}_i : \Lambda^i \mapsto \Lambda^{i+1}$ and the Hodge star operator by $*_i : \Lambda^i \mapsto \Lambda^{n-i}$. The 0-forms are functions on the manifold and 1-forms are discussed above. In the case of differential forms on a surface, all 2-forms can be represented as products of a function and the volume form. Using the Riemannian metric, we can additionally get a one-to-one correspondence between vector fields and 1-forms: to any vector field $\mathbf{v}$, we associate the 1-form $\langle \mathbf{v}, \cdot \rangle$. With these identifications of functions and vector fields with the 0, 1 and 2-forms on a surface, the operators on spaces of functions and vector fields can be expressed in terms of the exterior derivative and the Hodge star:

Fields	J	grad	curl	div
Forms	$*_1$	d_0	$*_2 \circ \mathrm{d}_1$	$*_2 \circ \mathrm{d}_1 \circ *_1$

In the same spirit, the Hodge decomposition, which is discussed for vector fields above, can alternatively be formulated for 1-forms.

4. Discretization

In most applications, directional fields are computed by solving an optimization problem, where the optimization variables depend on how the fields are represented and discretized. The choice of representation and discretization directly affects the properties of the optimization problem, such as linearity or convexity. Hence, these choices heavily influence the range of objectives and constraints that one can pose, and, as a result, determine which applications are computationally feasible and which are not.

In this section, we discuss various discretizations of tangent spaces and spaces of vector fields on triangle meshes. In addition, we present geometric and topological discretization challenges: the need to define a discrete connection between tangent spaces, and the ambiguities that arise due to the sampling process. The issues addressed here are the foundation for the directional field representations described in Section 5.

4.1. Tangent Spaces

The tangent spaces of the a triangle mesh can be located on the faces, edges, or vertices of a triangle mesh.

One way to construct a tangent space at a point is to assign a surface normal vector to the point. The tangent space is then the linear subspace of $\mathbb{R}^3$ orthogonal to the normal vector. For points inside the faces, the surface normal is obviously the normal of the triangle. Different schemes for computing surface normal vectors at the edges and vertices of a mesh have been proposed. Among those are weighted averages of the adjacent triangle normal vectors [Max99], and techniques based on principal component analysis [GH97].

As an alternative to this extrinsic construction, tangent vectors at a point on a mesh can be considered as the set of vectors pointing from the point along the surface. For example, the tangent vectors at a vertex point from the vertex into one of the neighboring triangles. This construction is typically used for working intrinsically on a surface, e.g., when shooting curves on a surface [PS98]. In this case, the tangent space at a vertex is the set of all possible vectors which are the tangent to all possible curves passing through this vertex. Intrinsic notions of tangent vectors have also been used in [ZMT06, KCPS13, MPZ14], where a smooth atlas is defined on the mesh using a local parametrization of the 1-ring of each vertex.

Note that the choice between these options is not just a matter of personal taste; it has consequences that can influence the suitability for specific use cases. A prominent example is the positioning of the singularities of a directional field. In most discrete field representations, they lie in between the tangent spaces, i.e. in the vertex-based scenario within the triangles, in the triangle-based scenario on the vertices. This implies that it may or may not be possible to position singularities onto sharp features of non-smooth surfaces.

4.2. Spaces of Vector Fields

Given a choice of discrete tangent spaces, we still need to fix the space of discrete tangent vector fields. While this choice is less discussed in the literature than the choice of representation, it is similarly important. Furthermore, for applications such as surface parametrization, where the main goal is to compute scalar functions, the choice of space for vector fields is closely tied to the choice of space for scalar functions.

A large portion of the literature in geometry processing uses scalar functions whose values are given at vertices and interpolated linearly to the faces. This space of functions is known as the linear *Lagrange elements*, and we denote it by S_h [PP03, War06] (see Figure 3). The gradients of such functions are vector fields which are constant at each face, and tangent to the faces. We denote this space by $\mathcal{X}_h$. Hence, if $f \in S_h$, then $\operatorname{grad} f \in \mathcal{X}_h$. In this sense, these two spaces fit together, and this combination is indeed common in the literature. In order to define discrete operators of vector calculus which are consistent with the smooth case (e.g. allow for a discrete Hodge decomposition as discussed in Section 7.1), it is useful to define another space of scalar functions that is linear across faces, albeit only continuous at edge midpoints. This space is known as the *Crouzeix-Raviart elements* [WBH*07], and denoted by S_h^* (see Figure 3). Gradients of functions in S_h^* are also piecewise constant vector fields on the faces.

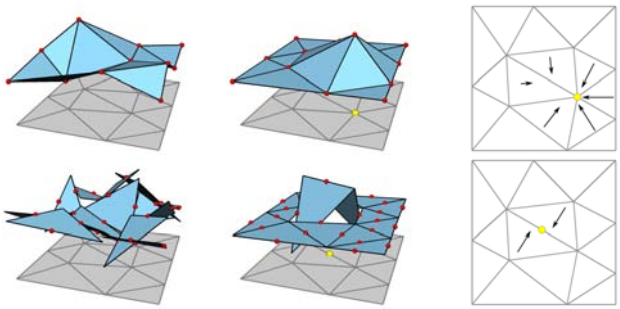


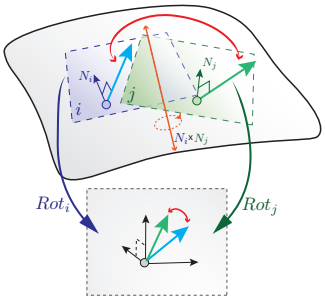
Figure 3: Graphs of functions in S_h (top row) and S_h^* (bottom row) and their gradients are shown. Image courtesy of Matthias Nieser.

Piecewise constant vector fields are discontinuous along edges. This can be problematic, depending on the application. Alternatively, higher order representations [ZMT06, KCPS13] can be used for constructing spaces of vector fields. Using a higher order interpolation scheme, for instance, allows to represent fields in such a way that integral lines do not intersect [RS14, MPZ14], properly preserving this property from the continuous setting, though recent results show that this is possible to certain extents with piecewise constant fields as well [CSZ16].

Spaces of vector fields are closely related to spaces of differential forms. For the construction of spaces of differential forms in *Discrete Exterior Calculus* and the duality of spaces of forms and vector fields, we refer the reader to [DHLM05].

4.3. Discrete Connections

Given two adjacent tangent spaces i and j, we need a notion of *connection* between them in order to compare two directional objects that are defined on them. In Section 3.1, it is explained that connections are tightly linked to the parallel transport of vectors. In the discrete setting, we describe a connection by specifying bijective linear maps between each pair of adjacent tangent spaces. We can think of the linear maps as the maps we obtain by parallel transport between the adjacent tangent spaces. In the case that the connection respects the metric of the surface (see Section 3.1), all maps between adjacent tangent spaces would be isometries. For a background on discrete connections, we refer to [KP15].



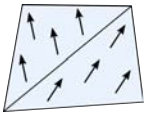
A straightforward discretization of the Levi-Civita connection is made by "flattening" the two adjacent tangent planes. Specifically, this is done by rotating them around the axis which is perpendicular to both their normals (the orange line in the inset) so that they coincide. The directionals in the rotated tangent spaces can then be compared directly, as they lie in the same Euclidean space. As a consequence, this process yields a three-dimensional rotation operator which allows to parallel-transport a vector from one tangent space to another. It is important to note that this definition of discrete

connection depends only on the normals. It is invariant to any local coordinate system, or to the specific representation of the directionals. Such a connection is required regardless of the choice of vector space: see e.g. [CDS10] for piecewise constant vector fields, and [KCPS13] for piecewise linear ones.

4.4. Discrete Field Topology

Moving from a continuous tangent bundle to a discrete set of tangent spaces, and from a continuous directional field to directionals per tangent space, can be considered a form of sampling. This sampling can lead to a loss of information, and introduce ambiguity. This, in particular, concerns the field topology (cf. Section 3.2), and is best exemplified as follows:

Consider a piecewise constant face-based 1-direction field that is discontinuous across the edges. As a consequence, the notions of smooth holonomy and index do not immediately apply in this case, e.g. the differential is not defined on the discontinuous edges. In order to extend these notions to the discrete case, the behavior of the field across the edges, where the field is discontinuous, needs to be clarified. The example in the inset shows such a piecewise constant field in two triangles i and j. In this example, it is intuitive to assume that the field undergoes a rotation $\delta_{ij} = \pi/4$ clockwise when crossing the common edge from top (i) to bottom (j). However, every other rotation $\delta_{ij} = \pi/4 + 2\pi k$, with $k \in \mathbb{Z}$, would be a valid assumption as well.

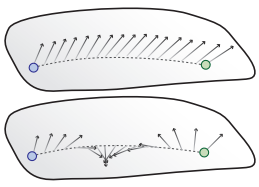


Rotation If no additional information is given, the reasonable assumption is that the field undergoes a *principal rotation* across a discontinuity. That means that the rotation δ_{ij} between two vectors $\mathbf{v_i}, \mathbf{v_j}$, in respective adjacent tangent planes i, j, is assumed to be in the range $[-\pi, \pi)$, measured following a parallel transport $i \to j$. In this case, the topology of the field is implicit, induced by the underlying assumption. If we indeed sample a continuous field, this implicit topology might of course differ from the original topology. This is an *aliasing* problem, analogous to similar problems in signal processing, where the sampling density is too sparse to capture the bandwidth of the signal. We discuss this in more detail in Section 6.

It is important to note that this problem is not limited to piecewise-constant vector fields, but that it exists in other forms of field sampling and interpolation as well. For instance, there are no discontinuities across edges in piecewise linear fields. However, the field is interpolated within each triangle, and this interpolation is subject to such aliasing artifacts as well.

It is, nevertheless, possible to achieve a higher power of expression for directional field interpolation and topology in the discrete setting; it is done by *explicitly* specifying the topology, or rather the rotations across discontinuities, as detailed in the following.

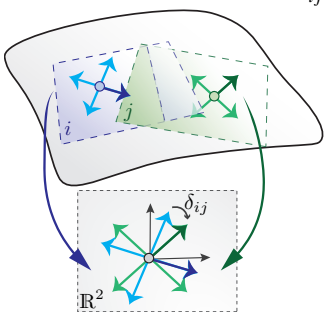
Period Jumps By specifying a value $k \in \mathbb{Z}$ (for each pair of adjacent tangent spaces), we can explicitly prescribe a non-principal rotation, which differs by k full period rotations (i.e., rotations by $2\pi k$ radians) from the principal rotation. This is shown in the inset for $k = 0$ and $k = 1$). This concept was introduced by Li et al. [LVRL06], where these values were denoted as *period jumps*. This way, the topology of the field can be controlled explicitly, and any field topology can be faithfully sampled, if the mesh resolution permits.



Explicit control over the period jumps can also be achieved by directly controlling the rotations δ. Note, however, that these rotations need to meet certain conditions to actually be consistent with a directional field [CDS10], as we detail in 5.1.

Matching If the field is multi-valued, with $N > 1$ directionals per tangent space, an additional degree of freedom needs to be taken into account: the correspondence between the individual directionals in tangent space i to those in the adjacent tangent space j. A *matching* between two N-sets of directionals: $\{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$ in tangent space i, and resp. directionals $\mathbf{v}$ in tangent space j, is a bijective map f between them (or their indices). Assuming an indexing of directionals within each tangent space in a counterclockwise order, the matching *preserves order* if and only if $f(\mathbf{u}_r) = \mathbf{v}_s \Leftrightarrow f(\mathbf{u}_{r+1}) = \mathbf{v}_{s+1}$ for all $0 \le r, s < N$, where the indices are taken modulo N. The term *matching* is generally meant to refer to a matching that preserves order, unless it is explicitly stated that it is *non-order-preserving*.

Effort Based on a matching f, the notions of rotation and principal rotation can be generalized to multi-valued fields. The rotation δ^r_{ij} of an individual directional $\mathbf{u}_r$ to its matched adjacent directional $\mathbf{v}_{f(r)}$ is defined just like the rotation of a 1-directional field. The sum $Y_{ij} = \sum_{r=1}^{N} \delta^r_{ij}$ is called the *effort* of the matching f. The value $\delta_{ij} = Y_{ij}/N$ is the average rotation, or simply *rotation* of the matching. Note that for a symmetric N-directional field $\delta_{ij} = \delta^r_{ij}$ for every r.



The efforts of different (order-preserving) matchings differ by 2π (regardless of N and of symmetries); the one matching for which $Y_{ij} \in [-\pi, \pi)$ or, equivalently, $\delta_{ij} \in [-\pi/N, \pi/N)$, is called the *principal matching*, and the corresponding rotation δ_{ij}, as in the 1-directional case, is called the *principal rotation*.

5. Representation

Unlike scalar functions, which can be unambiguously represented using a single number per point, directional data poses some challenges. The need to utilize different types of directional information has motivated many different representations. In the following, we describe the representations of directional fields that have been proposed in the literature.

5.1. Angle-Based

1-Direction Fields By defining an arbitrary reference base orthonormal frame $\{e_1, e_2\}$ on each tangent space, 1-direction (unit vector) fields can be concisely described within each tangent space by a signed angle ϕ that is relative to e_1 [LVRL06,RVLL08]. Following the Levi-Civita connection (flattening rotation; see Section 4.3), an extra angle X_{ij}, identified with the change of bases e between the

flattened tangent spaces i and j, is required. Finally, an integer k_{ij} describes a possible period jump between the directions. Having all these, the rotation angle between two adjacent 1-directionals ϕ_i and ϕ_j is expressed as follows:

$$\delta_{ij} = \phi_j - (\phi_i + X_{ij} + 2\pi k_{ij}). \tag{6}$$

The two directions ϕ_i, ϕ_j are parallel if δ_{ij} is zero.

Remark: in this representation, a principal rotation $\delta_{ij} \in [-\pi, \pi)$ is not generally associated with a vanishing period jump $k_{ij} = 0$ (cf. Section 4.4). That is because ϕ and X are not necessarily principal by definition. Period jumps between adjacent local frames are unavoidable, as they are vector fields that are subjected to the Poincaré-Hopf theorem as well.

N-Direction Fields Another advantage of the angle-based representation is the straightforward extension to N-direction. This is done in [LVRL06, RVLL08] by using a single direction ϕ representing the set of N directions, and allowing the period jump to be an integer multiple of $1/N$, thereby also enumerating the efforts of (order-preserving) matchings. The set of N directions $\{\phi + l \cdot 2\pi/N \mid l \in \{0 \dots N-1\}\}$ is trivially deduced by the N-symmetry from ϕ. The rotation angle formula becomes:

$$\delta_{ij} = \phi_j - (\phi_i + X_{ij} + \frac{2\pi}{N} k_{ij}) \tag{7}$$

Rotation Angles Instead of constructing local bases and expressing relative angles within each tangent space, it is possible to describe a field explicitly by the rotation angles δ_{ij} of the field between tangent spaces i and j. This is the representation used in [CDS10]. Note that this representation does not require the choice of local bases—the field is represented explicitly only in a single tangent space; the rest of the field is deduced by propagating the explicit rotations δ_{ij}. Note, however, that this representation is not inherently valid: the rotations must meet a consistency condition for every cycle of tangent spaces; otherwise, they are not consistently "integrable" to actual directionals per tangent space (cf. Section 11.2.1).

2^2-Direction Fields A particular angle-based representation was devised for frame-fields in [LXW*11]. For two independent 2-direction fields, represented by angles ϕ, ψ per face, the matching is represented by an extra binary switch variable: $q \in \{0, 1\}$ that encodes the two potential matchings between neighboring frames $(\phi, \psi)_i$ and $(\phi, \psi)_j$. We obtain two different rotation angles:

$$\delta(\phi, \psi)_{i \to j} = \begin{pmatrix} \phi_j - [(1-q)\phi_i + q\psi_i + X_{ij} + \pi k_{1,ij}] \\ \psi_j - [(1-q)\psi_i + q\phi_i + X_{ij} + \pi k_{2,ij}] \end{pmatrix} \tag{8}$$

An alternative angle-based representation for 2^2-direction fields is offered in [IBB15], which requires only a single period jump and can be seen as a direct generalization of [RVLL08, BZK09]. The key idea is to locally decompose a 2^2-direction (ϕ, ψ) into a 4-direction represented by $\alpha \in \mathbb{R}$, and an additional skew angle $\beta \in (-\frac{\pi}{4}, \frac{\pi}{4})$. The explicit relation w.r.t. the previous approach is given by $\phi = \alpha + \beta$ and $\psi = \alpha + \frac{\pi}{2} - \beta$, and an ordering of $[\phi, \psi, \phi + \pi, \psi + \pi]$ is assumed. Consequently, the rotation angles are fully determined by a single period jump as in the N-direction field case:

$$\delta(\phi, \psi)_{i \to j} = \begin{pmatrix} (\alpha_j + \beta_j) - (\alpha_i + (-1)^{k_{ij}} \beta_i + k_{ij} \frac{\pi}{2}) \\ (\alpha_j - \beta_j) - (\alpha_i - (-1)^{k_{ij}} \beta_i + k_{ij} \frac{\pi}{2}) \end{pmatrix} \tag{9}$$

The major advantage of the angle-based representation is that directions, as well as possible period jumps, are represented explicitly. This leads to a linear expression of the rotation angle as well. This is beneficial for optimization purposes (cf. Section 8). Moreover, this representation provides control over the topological properties of the field. The major disadvantage is the use of integer (and possibly binary [LXW*11]) variables, which leads to discrete optimization problems, as we discuss in Section 11.

5.2. Cartesian and Complex

A vector $\mathbf{v}$ in a two-dimensional tangent space can be represented using Cartesian coordinates (from $\mathbb{R}^2$) in the local coordinate system $\{e_1, e_2\}$, or equivalently as complex numbers (from $\mathbb{C}$) [KCPS13]. This representation is related to the angle-based representation via trigonometric functions, or the complex exponential, as follows:

$$\mathbf{v} = \begin{pmatrix} \cos(\phi) \\ \sin(\phi) \end{pmatrix} = e^{i\phi} \tag{10}$$

The change of bases from one tangent space to another, by angle X_{ij} as before, is performed via multiplication with a rotation matrix:

$$\begin{pmatrix} \cos(X_{ij}) & -\sin(X_{ij}) \\ \sin(X_{ij}) & \cos(X_{ij}) \end{pmatrix} \tag{11}$$

or, in complex notation, $e^{iX_{ij}}$. Note, however, that due to the periodicity of the trigonometric functions or the complex exponential, the Cartesian representation is invariant to rotations by multiples of 2π, and thus the rotation is inferred implicitly. When comparing adjacent vectors in this representation, the inferred rotation is then inherently principal (cf. Section 4.4). The lack of explicit period jumps can be an advantage, as such discrete jumps typically make optimization problems non-convex. However, it can be a disadvantage as well, as it is not possible to exert full control over the topology of the field (cf. Section 6).

By multiplying the argument of the trigonometric functions, or taking the complex exponential to the power of N, i.e. using

$$\mathbf{v}^N = \begin{pmatrix} \cos(N\phi) \\ \sin(N\phi) \end{pmatrix} = e^{iN\phi} \tag{12}$$

we achieve invariance to rotations by multiples of $2\pi/N$ instead of 2π. In this way, the principal matching for N-directional fields is implicitly assumed. This idea was introduced several times for the representation of N-direction fields under varying names [HZ00, RLL*06, PZ07, ZHT07, KLF13, KCPS13]. Parallel transport of these vectors is performed using the rotation matrix

$$\begin{pmatrix} \cos(NX_{ij}) & -\sin(NX_{ij}) \\ \sin(NX_{ij}) & \cos(NX_{ij}) \end{pmatrix} \tag{13}$$

or the complex number $e^{iNX_{ij}}$, respectively. The inferred principal rotation of this representative is exactly the principal matching of the represented N-direction field.

The Cartesian representation, unless explicitly constrained to unit vectors, can also represent N-vector (non-unit) fields with the magnitude component. This can be an advantage or disadvantage, depending on the use case and the optimizations to be performed (cf. Section 11).

5.3. Tensors

Tensors of rank 2 naturally show up in various contexts. Notable examples are curvature, metric, strain and stress. Tensors on a 2-manifold can be simply represented by real-valued 2x2 matrices in local coordinates:

$$T = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}$$

Tensor fields are an intensively studied topic; a comprehensive overview of the latest developments can be found in [LV12]. Symmetric tensors with $T_{12} = T_{21}$ are the most commonly used [ZHT07], due to their straightforward relation to directional information, as outlined in the following.

Symmetric Tensors A symmetric matrix $T \in \mathbb{R}^{2\times 2}$ has an eigendecomposition $T = U\Lambda U^T$ by definition, where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2)$ contains the two real eigenvalues, and $U = [u_1, u_2]$ contains the two (orthogonal) eigenvectors with $||u_i|| = 1$. A rank-2 tensor field encodes two eigenvector fields u_1 and u_2 accordingly. Since eigenvectors are only determined up to sign, a rank-2 tensor field can in fact be interpreted as two orthogonal 2-direction fields $\pm u_i$.

It is important to note that the eigendecomposition is unique only in case if $\lambda_1 \neq \lambda_2$. Thus, the directional information is solely contained in the traceless deviatoric part $D = T - \frac{tr(T)}{2} I$. Consequently, the eigenvalues λ_i' of D encode a 2-vector field $\pm\lambda_i' u_i$ [dGDT15] with singularities at points where D vanishes. A common approach to 2-tensor fields is to choose local coordinate systems (e.g. per face or per vertex), and to handle the connection discretely by transformations between such local coordinate systems. A well-founded alternative, using a coordinate-free representation, has been recently proposed in [dGLB*14], using discrete differential forms. The idea is to decompose a symmetric 2-tensor field in such a way that it can be stored as one scalar per vertex, edge and face of a triangulation, i.e., three discrete forms of orders 0, 1 and 2, respectively. This representation is consistent with discrete exterior calculus, as it allows the definition of discrete notions, among which are divergence-free tensors, covariant derivatives, and the Lie bracket.

Symmetric tensor fields should not be confused with orthogonal cross-fields (4-vector or direction fields). Although seemingly similar, they differ significantly in the class of possible singularities. While cross-fields may have singularities of type $k \cdot \frac{1}{4}$, $k \in \mathbb{Z}$, tensor fields are limited to singularities of type $k \cdot \frac{1}{2}$. Intuitively, this means that a symmetric tensor field can be unambiguously decomposed into two independent 2-direction fields, which is not possible for a general 4-direction field. This observation uncovers the limitations of tensor fields: while such fields are commonly used to estimate a sparse set of constraints (e.g. salient principal curvature directions), 4-direction fields are interpolated in the rest of the domain (cf. [BZK09]). For example, the smoothest 4-direction field on a cube, having eight singularities of index $\frac{1}{4}$ at its corners, does not correspond to a smooth symmetric tensor field.

Structure Tensors A special case of symmetric tensors are *structure tensors*, which are frequently used to represent 2-direction fields in arbitrary dimensions [GK95]. The key idea is to represent a line l in $\mathbb{R}^n$ as the eigenspace of the largest eigenvector of an $n \times n$ matrix

$$O_l = \frac{vv^{\mathsf{T}}}{||v||^2},$$

where v is a vector parallel to l. It is easy to see that the construction is invariant to flips or changes in magnitude in v or O_l, and that it uniquely identifies a given line l.

General Tensors General (not necessarily symmetric) tensors are not guaranteed to have real eigenvalues. Nevertheless, it is possible to deduce consistent directional information using the concept of dual eigenvectors [ZP05, ZYLL09] that applies to operators with complex eigenvalues. As a result, the directional field is partitioned into real and complex parts that smoothly join along separation curves. The field is defined by eigenvectors in the real parts, while dual-eigenvectors are chosen in the complex parts.

5.4. Composite

A 2^2-vector field can be represented as a composition of a 4-direction field and an additional auxiliary field that encode scale and skewness, i.e. the difference of the 2^2-vector field from the 4-direction field. [PPTSH14] proposes a unique decomposition into a 4-direction field and a 2×2 semi-positive definite tensor field W. This representation completely decouples the scale and skewness of a 2^2-vector from the direction component, allowing to interpolate them independently. In particular, since the tensor is semi-positive definite, it can be efficiently interpolated coefficient-wise by simply solving a sparse linear system. In [JFH*15], a new Riemannian metric is computed on the surface. This metric serves as the semi-positive definite tensor that defines the skewness and scale. Following this, the 4-direction field is computed in this new metric.

5.5. 1-Forms

Instead of directly representing vector fields, it is possible to take the dual perspective, and represent 1-forms. A discrete 1-form is encoded using the integral of a continuous 1-form over the edges. Hence, for every oriented edge e_{ij} we encode a scalar $c_{ij} = \int_{e_{ij}} \omega_p(\hat{e}_{ij}) ds$, where $\hat{e}_{ij}$ is the corresponding unit vector, and p is a point along the edge [Hir03, FSDH07].

The advantage of this approach is that we are encoding scalar values (the integrated projection of the vector field along the edges), and therefore the approach is coordinate free (i.e. no local frame is required). In addition, there are discrete versions for the curl and the divergence of a vector field represented as a 1-form, which are simple to represent and manipulate (using the corresponding exterior calculus operators d and $\star$). Moreover, the space of discrete 1-forms admits a discrete Hodge decomposition [FSDH07]. See also Section 7.1. Discrete 1-forms, encoded on edges, can be extended into the adjacent triangles using Whitney forms [WTD*06]. This allows for finite element discretizations of exterior calculus [AFW06].

Discrete 1-forms have been successfully used in applications such as vector field design [WTD*06, FSDH07, BCBSG10], quad meshing [TACSD06], point cloud meshing [TGG06] and surface parametrization [GY02, GGT06]. In addition, there is a large body of work specifically addressing harmonic 1-forms and discrete analytic functions (see e.g. [Mer01]).

5.6. Complex Polynomials

Complex Cartesian representations have been extended to general 1^N-vector fields in [DVPSH14]. The main insight is that the Cartesian representation $\mathbf{u}^N$ is in fact equivalent to the root-set of the complex polynomial $p(z) = z^N - \mathbf{u}^N$. Analogously, every N-vector set $\{\mathbf{u}_1, \dots \mathbf{u}_N\}$, in the complex form $\mathbf{u_i} \in \mathbb{C}$, can be uniquely identified as the roots of a complex polynomial $p(z) = (z-\mathbf{u}_1)\dots(z-\mathbf{u}_N)$. Writing p in monomial form, $p(z) = \sum_i \mathbf{c_n} z^n$, the coefficient set $\{\mathbf{c_n}\}$ is thus an order-invariant representative of a 1^N-vector, and was denoted as an *N-PolyVector*. Comparing between PolyVectors on adjacent tangent spaces amounts to comparing polynomial coefficients. As every coefficient $\mathbf{c_n}$ contains multiplications of $N-n$ roots, the coefficients are compared accordingly: assume two coefficients sets $\mathbf{c_n}$ and $\mathbf{d_n}$ on neighbouring tangent spaces, then

$$\mathbf{d_n} = e^{i(N-n)\delta_{n,ij}} \mathbf{c_n}, \tag{14}$$

Where $e^{i(N-n)\delta_{n,ij}}$ is the rotation between the coefficients $\mathbf{c_n}$ and $\mathbf{d_n}$ (including the change of basis X_{ij}).

The advantages (continuous optimization) and disadvantages (coupling of magnitude and direction) of complex Cartesian representations are inherited by the PolyVector representation, though it represents the general case of 1^N-vector fields. Moreover, an N-vector is represented by an N-PolyVector in which all the coefficients but the free coefficient, $\mathbf{c_0}$, are zero. This constitutes a simple linear subspace. In this manner, the rest of the coefficients represent the skewness of the 1^N-vector—in an implicit manner with no obvious way to control it. Furthermore, it was shown in [DVPSH14] that the effort of any matching between PolyVectors in adjacent tangent spaces is equal to the effort of a respective matching between their free coefficients $\mathbf{c_0}$ and $\mathbf{d_0}$, when considered as the higher-order complex representatives of an underlying N-direction, as in Section 5.2. Thus, the concept of principal effort and matching readily applies to PolyVectors as well.

5.7. Linear Operators

In the continuous case, given a vector field $\mathbf{v}$, we can construct a linear operator from functions on $\mathcal{M}$ to functions on $\mathcal{M}$ given by: $f \mapsto \langle \text{grad}\, f, \mathbf{v} \rangle$. In fact, the opposite is also true: given a linear operator on functions which fulfills the product rule, it is possible to construct the *unique* corresponding vector field [Mor01]. Therefore, one possible representation of vector fields is through their correponding linear operators on functions.

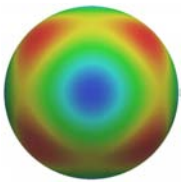
By choosing a discrete function space, for example S_h, this linear operator can be represented as a sparse matrix in the discrete setting. Alternatively, one can choose a set of k lowest eigenfunctions of the Laplace-Beltrami operator as the function space, leading to a small $k \times k$ matrix representation.

This point of view allows to design vector fields under various global constraints (such as commutativity with a symmetry map), as well as combining such constraints with point-wise constraints on the value of the vector field [ABCCO13]. In addition, this approach allows to compute the *flow* of a vector field using the exponential of the matrix representing the operator, which is useful for numerical fluid simulation [AWO*14], and for generating smooth maps between surfaces [COC15].

An important disadvantage of this approach is that the product rule $D_{\mathbf{v}}(fg) = fD_{\mathbf{v}}g + gD_{\mathbf{v}}f$ does not hold in the discrete case. Hence, given a linear operator it is challenging to check whether it corresponds to a vector field without projecting on the chosen basis. This projection could potentially be a costly operation.

5.8. Spherical Harmonics

The Cartesian or complex coordinates (cf. Section 5.2) can be interpreted as coefficients of a certain class of 2D spherical harmonics [RS15]. The directions of an N-direction field then correspond to the maxima of the function described by these coefficients. This interpretation is useful because it can be generalized to 3D [HTWB11]. The inset shows a visualization of a function from the employed class of spherical harmonics. Note that there are six maxima (blue), representing six directions forming an orthogonal 3D cross. Comparison and interpolation of 3D crosses can then be reduced to interpolation of coefficients.



It is important to note that the space of functions that represent rotated crosses is a proper submanifold of the full coefficient space. While for 2D crosses the projection on the 1D submanifold of rotations turns out to be a simple re-normalization, the corresponding operation for 3D crosses is considerably more involved. The projection from the 9D space of free SPH coefficients onto the 3D submanifold of functions, associated with rotated crosses, is highly nonlinear, and globally-optimal schemes [KCPS13] do not generalize to 3D.

5.9. Scalar Fields

Gradient vector fields of scalar fields are inherently curl-free, and co-gradient vector fields are inherently divergence-free. Hence, a convenient option to represent and synthesize such fields without the need for respective constraints is to deal with scalar fields instead, and derive the vector field in the end. For instance, in [vFTS06], a divergence-free 2D vector field is represented as the co-gradient field of a scalar field. Similarly, a divergence-free 3D vector field can be defined as the cross-product of the gradients of two volumetric scalar functions [vFTS06]. This representation has also been used to encode a 2-vector field in [YCLJ12], where the field is represented as the set of directions perpendicular to the gradient of a scalar function.

6. Topology

The discrete Levi-Civita connection, the principal rotations, the period jumps, and the matchings that are part of the various directional field representations bring about discrete counterparts of curvature and singularity indices. We describe how these are acquired in the different representations.

6.1. Direction Fields and Trivial Connections

Given a direction in one of the tangent spaces, one might try to create a complete direction field by propagating the direction, through parallel transport, into all other tangent spaces. This is inconsistent

in the presence of curvature, as then the Levi-Civita connection along a cycle does not transport a direction back to itself due to holonomy, cf. Section 3.1.

By using an alternative transport with zero holonomy, however, a direction field is well-defined in the above manner. A connection with this property is called *trivial* [CDS10]. We can thus look at a given direction field (with matchings between adjacent tangent spaces) as an "altered connection", where the rotation angles δ_{ij} of the field (cf. Section 4.4) describe the deviation from the Levi-Civita connection. The connection implied by an N-direction field is trivial in the sense that a directional is always transported back to itself up to a rotation by $k\frac{2\pi}{N}, k \in \mathbb{Z}$.

6.2. Singularities and Indices

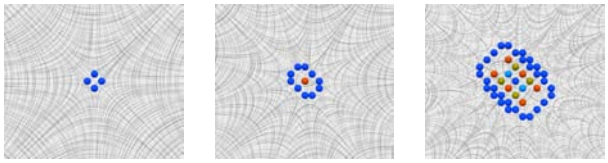
The singularities of a directional field (cf. Section 3.2) are a topological property that is derived, in the discrete setting, from the sums of rotations δ_{ij} around elementary cycles of tangent spaces. For instance, in the case of a face-based field representation, the cycles are 1-rings around vertices. In the case of a vertex-based representation, the cycles comprise the edges bounding a face. The sum $D_{\mathcal{C}} = \sum_i \delta_{i,i+1}$ along such a cycle $\mathcal{C}$ is the difference between the curvature $K'_{\mathcal{C}}$, induced from the trivial connection defined by the field, and the original Gaussian curvature $K_{\mathcal{C}}$, induced by the Levi-Civita connection: $K'_{\mathcal{C}} = K_{\mathcal{C}} + D_{\mathcal{C}}$. Note that in a face-based field representation, $K_{\mathcal{C}}$ is simply the angle-defect from 2π [CDS10] at the vertex enclosed by $\mathcal{C}$.

Wherever $K'_{\mathcal{C}} \neq 0$, the field has a singularity, and its index is $K'_{\mathcal{C}}/2\pi$ [RVAL09, CDS10]. There are several, yet equivalent, expressions for the calculation of singularity indices [LVRL06, RVLL08, DVPSH14].

Surfaces that are not simply-connected admit non-contractible (and boundary) cycles, forming a homology basis [RVLL08, CDS10]. In addition to the rotation sums around elementary (e.g., 1-ring) cycles, the rotation sums around these non-elementary cycles are additional topological degrees of freedom of the field.

6.3. Sampling Problem

We discussed the ambiguities that a discrete field representation introduces in Section 4.4. Such ambiguities can be settled using explicit period jumps and matchings (e.g. in an angle-based representation, cf. Section 5.1). In representations that do not natively carry this extra information (e.g. the Cartesian or complex representation), one can only implicitly assume principal matchings and rotations (cf. Section 4.4)—unless some other prior information about the singularities is available. For the case of an 1^N-direction field, this means that the rotation between two adjacent tangent spaces is always only within $[-\pi/N, \pi/N)$. Consequently, the rotation sum around a cycle of m tangent spaces cannot exceed $m\pi/N$. Therefore, higher-order singularities cannot be represented by low-valence cycles. For instance, in a vertex-based representation on a triangle mesh (as in [KCPS13]), no other indices besides $\pm\frac{1}{N}$ are likely to arise. If the geometry or the constraints promote higher index, clusters of $\pm\frac{1}{N}$ singularities arise instead. Figure 4 shows examples of fields with a high-order singularity, sampled in a face based setting.



index $-\frac{4}{4}$ index $-\frac{9}{4}$ index -$\frac{21}{4}$

Figure 4: Using the principal period jumps and matchings in a sampling of a 4-direction field splits higher-order singularities into lower-order ones. The colored dots correspond to singularities of index $-\frac{2}{4}$ (light blue), $-\frac{1}{4}$ (blue), $\frac{1}{4}$ (red) and $\frac{2}{4}$ (gold).

7. Operators

7.1. Discrete Vector Calculus

We discuss discrete differential operators acting on spaces of piecewise constant vector fields and piecewise linear functions on triangular surface meshes, which were discussed in Section 4.2. We introduce conforming and nonconforming discrete divergence and curl operators and show how these operators can be combined to get a discrete Hodge decomposition of vector fields. The presented concepts have been introduced in [PP03, War06] and our presentation loosely follows theirs.

Alternatively to this construction, a structure-preserving discrete Hodge decomposition can be formulated in terms of discrete differential forms and discrete operators between them. For a general treatment of *Discrete Exterior Calculus*, we refer to [Hir03, DHLM05] and to [FSDH07] for the discrete Hodge decomposition of discrete one-forms. For an introduction to DEC, we refer to [CdGDS13].

Discrete Divergence and Curl For simplicity, we consider only a triangular surface mesh $\mathcal{M}_h$ without boundaries. The vector fields in $\mathcal{X}_h$ are not differentiable, hence the definition (2) of the divergence cannot be directly applied. However, the right-hand side of (3) is well-defined for pairs of a function f in S_h or S_h^* and a vector field $\mathbf{v}$ in $\mathcal{X}_h$. Hence, we can evaluate integrals over $f \operatorname{div} \mathbf{v}$ and use this for defining a conforming and nonconforming discrete divergence. For any $\mathbf{v} \in \mathcal{X}_h$, the conforming discrete divergence of $\mathbf{v}$ is the linear functional

$$\begin{aligned} \operatorname{div}_h \mathbf{v} : S_h &\mapsto \mathbb{R} \\ f &\to -\int_{\mathcal{M}_h} \langle \operatorname{grad} f, \mathbf{v} \rangle \, \mathrm{d}A, \end{aligned}$$

and the nonconforming discrete divergence is the linear functional

$$\begin{aligned} \operatorname{div}_h^* \mathbf{v} : S_h^* &\mapsto \mathbb{R} \\ g &\to -\int_{\mathcal{M}_h} \langle \operatorname{grad} g, \mathbf{v} \rangle \, \mathrm{d}A. \end{aligned}$$

Following the definition of the curl in the continuous case, see (3), the conforming and nonconforming discrete curl operators are defined as

$$\operatorname{curl}_h \mathbf{v} = -\operatorname{div}_h \mathrm{J}\,\mathbf{v} \quad \text{and} \quad \operatorname{curl}_h^* \mathbf{v} = -\operatorname{div}_h^* \mathrm{J}\,\mathbf{v}.$$

The discrete divergence and curl of a vector field are functionals and not functions. This means they cannot be evaluated at a point of the surface, but they can only be tested with a function. In this sense,

they are integrated and not pointwise quantities. Since the space S_h and S_h^* are finite-dimensional, the functionals can be transposed to get functions representing the divergence and curl of a piecewise constant vector field. This transposition means multiplication with the inverse mass matrix. For a discussion of integrated and pointwise quantities and their relation, we refer to [WBH*07].

Discrete Harmonic Vector Fields For the construction of the discrete harmonic vector fields, we consider the kernels of the discrete divergence and curl operators. By Kernel(div_h) we denote the subspace of $\mathcal{X}_h$ containing the vector fields $\mathbf{v}$ for that $\mathrm{div}_h\mathbf{v}$ is the trivial functional (i.e., $\mathrm{div}_h\mathbf{v}(f) = 0\ \forall f \in S_h$). The kernels of div_h^*, curl_h and curl_h^* are defined analogously.

To get spaces of harmonic fields whose dimension equals twice the genus of the surface, we need to combine the conforming and nonconforming discrete operators. There are two combinations: We define the *conforming discrete harmonic* vector fields as

$$\mathcal{H}_h = \mathrm{Kernel}(\mathrm{div}_h) \cap \mathrm{Kernel}(\mathrm{curl}_h^*)$$

and the *nonconforming discrete harmonic* vector fields as

$$\mathcal{H}_h^* = \mathrm{Kernel}(\mathrm{div}_h^*) \cap \mathrm{Kernel}(\mathrm{curl}_h).$$

We would like to remark that using only the conforming discrete divergence and curl yields a space of harmonic vector fields whose dimension is not twice the genus but depends explicitly on the number of vertices, edges and faces of the mesh.

Discrete Hodge Decomposition Similar to the definition of the discrete hamonic vector fields, the discrete Hodge decompositions combine the conforming and nonconforming discrete function spaces and operators. There are two possible combinations:

$$\mathcal{X}_h = \mathrm{Image}(\mathrm{grad}_{|S_h}) \oplus \mathrm{Image}(\mathrm{J}\,\mathrm{grad}_{|S_h^*}) \oplus \mathcal{H}_h$$

and

$$\mathcal{X}_h = \mathrm{Image}(\mathrm{grad}_{|S_h^*}) \oplus \mathrm{Image}(\mathrm{J}\,\mathrm{grad}_{|S_h}) \oplus \mathcal{H}_h^*.$$

As in the continuous case, the subspaces are mutally orthogonal with respect to the L^2-scalar product. The fact that there are two isomorphic decompositions is specific to the discrete setting and does not appear in the continuous case. In [War06], convergence of the decompositions under refinement has been established. In this sense, the two decompositions are similar as they converge to the same limit under refinement.

Discrete Killing Vector Fields Beyond scalar-valued derivative constraints such as the divergence and the curl, one can also pose more complex constraints, which can be expressed in terms of the covariant derivative tensor of the vector field. For example, one can consider the amount of *stretch* that a vector field generates: if we place two particles near each other on the surface and let them flow with the vector field, this stretch measures how much the distance between them changes while flowing. Vector fields that generate no stretch are called *Killing vector fields*, and are the generators of self-isometries (distance preserving maps from the surface to itself).

In terms of the covariant derivative, a vector field $\mathbf{v}$ is Killing if and only if its covariant derivative tensor is anti-symmetric, namely:

$$\langle \nabla_{\mathbf{u}}\mathbf{v}, \mathbf{w} \rangle = -\langle \nabla_{\mathbf{w}}\mathbf{v}, \mathbf{u} \rangle, \tag{15}$$

for any two vector fields $\mathbf{u}, \mathbf{v}$ (see [dC92], Chapter 3, Exercise 5). For example, it is easy to check that the planar vector field which rotates around the origin $\mathbf{v}(x,y) = (-y,x)$ has an anti-symmetric Jacobian matrix (the planar equivalent of the covariant derivative), and is therefore a Killing vector field. Exact Killing fields are quite rare, as their existence implies a 1-parameter family of self-isometries. For example, surfaces of revolution (and their isometric deformations) have exact an exact Killing field which generates the rotation around their axis. Therefore, in the discrete case it is interesting to consider *approximate Killing vector fields* (AKVFs), i.e. those whose flow is an approximate isometry.

Different discretizations have been used to compute approximate Killing vector fields. In [BCBSG10], the authors reformulate the Killing equation (15) in terms of exterior calculus, and use discrete 1-forms for the computation. As KVFs are generators of isometries, [GRK12] used Generalized Multi-Dimensional Scaling, a tool which has been previously used for computing approximate isometries between surfaces, for computing AKVFs. Another property of KVFs is that their corresponding derivation operator commutes with the Laplace-Beltrami operator. This property was leveraged in [ABCCO13], using the operator representation of vector fields, for computing AKVFs. Finally, [dGLB*14] used their general tensor decomposition, and [AOCBC15] used equation (15) by directly discretizing the covariant derivative.

As generators of isometries, AKVFs are useful in any application where distortion minimization is a goal. For example, AKVFs can be used for generating intrinsic patterns [BCBSG10], for segmentation [SBCBG11b] and for planar deformation [SBCBG11a].

Discrete Covariant Derivative While some differential quantities of vector fields have simple operators (for example, the divergence and the curl), in general, *any* first order differential operator can be expressed in terms of the covariant derivative tensor of the vector field. Thus, given a general discretization of the covariant derivative, we have more options for the type of objective functions we can minimize and constraints we can enforce. However, we need to pay for this flexibility by discretizing a more complicated object (a tensor instead of an operator from vector fields to functions, for example).

This direction is relatively new, and therefore only a few discrete constructions have been proposed so far in geometry processing. In [dGLB*14], the authors propose a general discretization of tensor fields on triangulations, using a decomposition which represents such tensors using five scalar functions. On the other hand, [AOCBC15] discretize directly the covariant derivative through its extrinsic representation as the directional derivative of the coordinates of the vector field, projected on the surface. Using both approaches it is possible to find approximate Killing vector fields by minimizing the norm of the symmetric part of the covariant derivative, compute the Lie bracket of two vector fields, as well as other applications.

8. Objectives and Constraints

Depending on the application, various objectives can be used for vector field optimization (Figure 5). We introduce the fairness objectives that have been proposed in the literature, and give an overview of the constraints that can be enforced while minimizing these objective functions.

8.1. Fairness

In order to optimize for the "best" direction field for a given application, the notion of "best" has to be defined and formulated. The most common way to measure the fairness of a field is by using the Dirichlet energy, measuring how variable, or rather, non-similar, the field is between adjacent tangent spaces. This notion poses several issues, which we show in the following.

Parallelity For many use cases, the ideal field is a parallel field, i.e. the direction in one tangent space is obtained via parallel transport from the directions in adjacent tangent spaces. Since globally parallel direction fields are not possible in the presence of curvature, many methods instead opt for "as-parallel-as-possible". In angle-based methods [RVAL09, CDS10] for N-direction fields, this amounts to a Dirichlet energy on the rotation angle

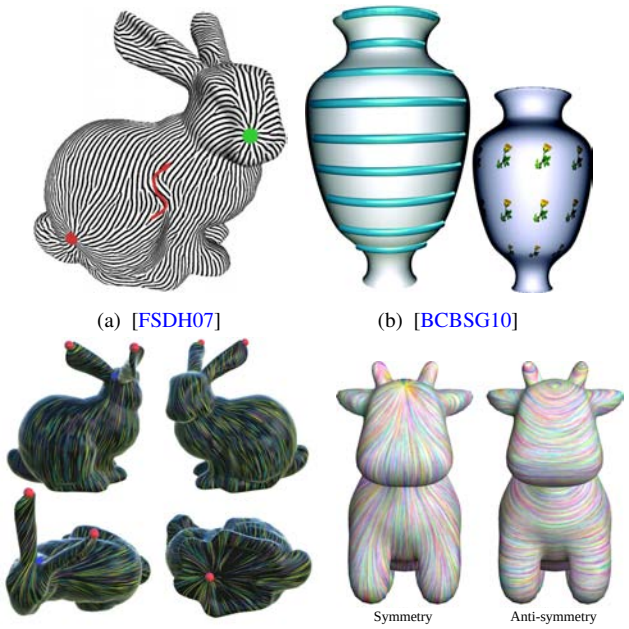
$$E_{\text{fair-}N} = \frac{N}{2} \sum_{e \in \mathcal{E}} w_e \cdot (\delta_e)^2, \tag{16}$$

where the edge weights w_e are chosen to account for a certain metric. Typical choices are unit weights [RVLL08, BZK09] or (dual) cotan weights [CDS10]. Following [IBB15], this fairness objective generalizes to 2^2-direction fields:

$$E_{\text{fair-}2^2} = \sum_{e \in \mathcal{E}} w_e \cdot (\delta(\phi, \psi)_e)^2. \tag{17}$$

Note that this objective can be decomposed into

$$E_{\text{fair-}2^2} = E_{\text{fair-4}} + E_{\text{fair-skew}},$$



(a) [FSDH07] (b) [BCBSG10]

(c) [KCPS13] (d) [ABCCO13]

Figure 5: Various objectives used for vector field optimization - (a) alignment to constraints, (b) Killing energy for isometric on-surface deformations, (c) smoothness (Dirichlet energy) (d) commutativity with the symmetry/antisymmetry self maps.

with a symmetric part that is the Dirichlet energy $E_{\text{fair-4}}$ of a 4-direction field, and a skew part corresponding to Eqn. (9):

$$E_{\text{fair-skew}} = \sum_{e_{ij} \in \mathcal{E}} w_{e_{ij}} \cdot \left((-1)^{p_{ij}} \beta_i - \beta_j\right)^2,$$

that measures the fairness of the specific component that encodes the deviation from orthogonality. This observation stresses the fact that a 2^2-vector field can be interpreted as a 4-direction field in a different metric [PPTSH14, JFH*15].

In the PolyVector representation [DVPSH14] (as the general case of the complex Cartesian representation [ZHT07, KCPS13]), the objective function is the difference between parallel-transported coefficients of the representing polynomial $p(z) = \sum_n C_n z^n$:

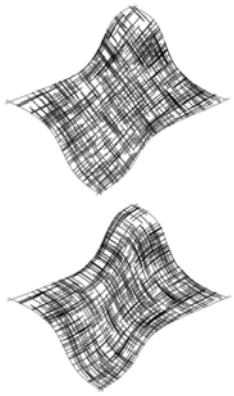
$$E_{\text{fair}} = \sum_{(i,j) \in \mathcal{E}} \sum_n \left| C_{n,j} - e^{i(N-n)X_{ij}} C_{n,i} \right|^2. \tag{18}$$

In the case of an entirely parallel field, all these (representation-dependent) objectives are zero; however, they behave differently in other cases. See Section 11 for a detailed discussion and comparison of these objective functions.

Note that a parallelity-based fairness objective strongly depends on the underlying connection for parallel transport. Variants exist that manipulate the usual Levi-Civita connection intrinsically [RVAL09] or extrinsically [ECBK14] in order to achieve certain effects, in particular preventing excessive singularities from arising in regions of high-frequency geometric detail or noise. This process can be interpreted as a re-distribution of Gaussian curvature, which accordingly influences where singularities arise.

Orthogonality Works that target 1^4-vector fields that are not necessarily symmetric 4-direction fields often attempt to make the field as orthogonal as possible (subject to other constraints), to avoid degenerate and small-angle configurations. In [LXW*11, IBB15], the relative angle deviation for the 2^2-direction fields from becoming 4-direction fields is directly encoded. [LXW*11] use it in an inequality bound to avoid degeneracy, while [IBB15] minimize the orthogonality deviation angle directly. In [DVPSH14], the angle is not directly encoded. Instead, 4-direction fields are represented by polynomials of the form $p(z) = z^4 - \mathbf{u}^4$, and thus minimizing the magnitude of all the coefficients of the polynomial, save for the free coefficient, provides control over orthogonality as well. Note that this trivially extends to controlling how much 1^N-vector fields are far from being N-direction fields.

Coons Interpolation As-parallel-as-possible objective functions promote straight streamlines. This might not always be a desired goal: in the context of surface reconstruction from sketches, [IBB15] estimate 3D normal fields from 2D concept sketches. The corresponding mathematical task is to find a 2^2-vector field in the image plane that is the projection of the (regularized) principal-curvature field of the corresponding (unknown) 3D shape. It turns out that an as-parallel-as-possible



optimization inappropriately flattens the sketched shape (inset, top) due to the metric distortion of the projection. In order to more naturally extrapolate the bending of sketched curves (boundary in the inset) to the shape, the theory of regularized curvature fields [IBB15] devises another optimization objective, called *bend field energy* that is based on the covariant derivative:

$$E_{bend2D} = \int ||\nabla_u \mathbf{v}||^2 + ||\nabla_v \mathbf{u}||^2 \, dA. \tag{19}$$

The corresponding 2^2-vector field is locally parametrized by vectors $\mathbf{u}$ and $\mathbf{v}$. Essentially, the objective function measures how smooth one vector field is in the direction of the other, and vice versa, instead of the isotropic smoothness of the parallelity measure. The result is an interpolation (inset, bottom) similar to Coons interpolation [FH99].

Soft Feature Alignment In certain applications, it is desirable to compute directional fields aligned with surface features. While this is achievable using constraints (Section 8.2), it can be accounted for directly in the fairness energy by measuring parallelity on the ambient space [JTPSH15]. This approach does not require any parameter tuning and it also does not require the explicit estimation of curvature directions. However, the generated fields are not guaranteed to follow all surface features.

Other Objective Functions A decomposition of the Dirichlet energy E_D into its holomorphic part E_H and its anti-holomorphic part E_A is proposed in [KCPS13]. This enables a new parameter $s \in [-1,1]$ to balance between both parts, i.e. $E(s) = (1+s)E_H + (1-s)E_A$. The authors observed that varying s can be useful for finding a good compromise between straightness of the field and number of singularities. Note that both parts of the Dirichlet energy have been used before: the holomorphic one in the context of Killing vector fields [BCBSG10], that are both holomorphic and divergence free, and the anti-holomorphic one for vector field design [FSDH07].

Some methods seek to minimize the curl of an existing (computed) direction field by scaling, in order to achieve better integrability. In [RLL*06], the curl is minimized isotropically on a 4-direction field, to form a 4-vector field with potentially less curl, leading to a more conformal than isometric parametrization. This is extended to different scaling in each axis to get an orthogonal and anisotropic, 2^2-vector field in [ZHLB10].

8.2. Constraints

Alignment It is often required for a field to be more than just fair or symmetric, but also to fit certain prescribed directions, or an entire existing field on a surface. Examples are the principal curvature 2^2 direction field, strokes given by an artist on the surface, boundary curves, or feature lines. If this information is represented in a way compatible with the employed field representation, this can be straightforward: the field can be either compared against the prescribed values, using a data term [KCPS13, DVPSH14], or be hard-constrained [BZK09]. Similar techniques are applicable when using differential forms [FSDH07] and functional operators [ABCCO13]. Difficulties arise when the constraints are *partial*, i.e., only one of multiple directions is prescribed at a point. Depending on the representation, such constraints can be hard to express. [IBB15] and [DVPSH15] recently described how such partial constraints can be handled for 2^2-directional fields. In the former case, due to the variable period jumps, it does not matter which of the four directions one constrains, i.e. without loss of generality one can simply constrain $\alpha+\beta$ (see Section 5.1) to a fixed direction. In [DVPSH15], partial constraints are encoded by lowering the degree of the complex polynomial. This is possible without loss of generality, since the order of roots does not matter (commutativity of multiplication).

Symmetry and Maps In addition to specifying exact directions (either sparse, full, or partial), it is possible to prescribe the behavior of the values of the vector field in a more global way. For example, if the surface has bilateral symmetry, it is advantageous if the designed directional fields adhere to the same symmetry, allowing field-guided applications to preserve the symmetry as well. Similarly, given multiple shapes with a correspondence between them, we could require that the directional fields commute with the correspondence, effectively designing directional fields *jointly* on multiple shapes.

In [PLPZ12], symmetric fields are computed by explicitly transporting the directional field using the symmetry map, which can be any self map of the surface. Using an alternative representation of a map as a correspondence between functions, [ABCCO13] compute symmetric (and anti-symmetric) vector fields by computing vector fields whose functional representation commutes (or anti-commutes) with the functional map. The latter generates smoother fields (as it is works in the space of smooth vector fields), while the former can be applied to more general setups, such as 4-direction fields.

8.3. Differential Constraints

Fields must fulfill certain differential properties in order to best suit their purpose. Curl-free fields are optimal for the purpose of integration to scalar fields or parametrization maps. Curl-free 1^N-vectors fields, with a focus on 2^2-vector fields, are described and computed in [DVPSH15]. Curl-free directional fields are defined by a reduction to a given matching, for which all adjacent matched vectors are curl free. This algorithm inherits the advantages and disadvantages of the PolyVector framework, and in addition tends to introduce many singularities in curved regions.

Divergence-free fields lead to volume preserving maps, and are therefore used in deformation [vFTS06], shape correspondence [COC15] and physically based simulation of fluids [AWO*14]. In $\mathbb{R}^3$, divergence-free fields can be represented as the cross product of the gradients of two scalar fields p,q: $\mathbf{u}(x,y,z) = \nabla p(x,y,z) \times \nabla q(x,y,z)$. On closed genus-0 surfaces, divergence-free fields are spanned by $\pi/2$-*rotated gradients* of scalar functions, see Section 7.1 and [PP03].

8.4. Topology

Representations that include explicit period jumps, like the angle-based methods discussed in Section 5.1, make it possible to control the topology of the field. This is achieved by prescribing rotation angle sums around 1-rings to fix singularities (cf. Section 6.2), or, more generally, by prescribing rotation angle sums around arbitrary cycles [RVLL08, BZK09, CDS10]. In Section 11, we discuss which methods support which types of such topological constraints.

9. Applications

Directional fields are very popular in computer graphics, scientific visualization, meshing for finite element simulations, cultural heritage and architectural geometry. In this section, we provide an overview of the most recent works that use direction fields, grouping them by applications.

9.1. Mesh Generation

Vector fields have been extensively used to provide directional guidance in automatic mesh generation methods [GE88]. For instance, they guide the mesh generation to the domain boundaries, and describe the characteristics of the governing equations of physical problems.

2D Knupp [Knu95] describes a method to generate quadrilateral meshes from curvilinear grids. These grids are obtained from solutions to a Poisson problem that is formulated with respect to a given vector field. Essentially, a global parametrization is sought, whose gradients match given vector fields as much as possible in a least-squares sense. Such early methods were restricted to planar domains and entirely regular quad grids.

Surfaces The same principle has been introduced to the field of Computer Graphics in a more general formulation by Ray et al. [RLL*06]. Parametrization-based mesh generation techniques, guided by vector fields, became quite popular recently [KNP07, BZK09, PTSZ11, LLZ*11, TPP*11, CBK12, NPPZ12, BCE*13, ECBK14, CK14b, CK14a, LLW15, JTPSH15]. The most significant difference between the early approaches and the more recent ones is the transition from vector fields to 4-direction ("cross") fields. This enables the handling of arbitrary geometries and topologies, and allows for the generation of "unstructured" quad meshes with irregular vertices, permitting higher flexibility and better quality. On surfaces, the direction fields are often used to promote alignment to the principal curvature directions of the underlying surface, through field-guided parametrization [RLL*06, KNP07] or field-guided mesh structure generation [CBK12, CHK13]. This is of interest [LRL06, ACSD*03, CSAD04, CK14b], e.g. to maximize approximation quality [D'A00], minimize normal noise and aliasing [BK01], or optimize element planarity [LXW*11]. Nevertheless, the fields serve an additional purpose to directional guidance in unstructured mesh generation; their role is actually two-fold: the *topology* of the fields is exploited to predetermine aspects of the mesh structure – in particular, the number, type, and position of irregular vertices are generally derived from the singularities in the fields – thereby greatly simplifying the parametrization optimization problems involved in the mesh generation process [KNP07, BZK09]. Anisotropic and adaptive meshing are also possible, if the guiding 4-direction field is replaced with a 2^2-direction ("frame") field [PPTSH14, DVPSH14, JFH*15, DVPSH15, CBK15].

Point Clouds Directional fields can be discretized and designed directly on point clouds [PTSZ11, LLZ*11, JTPSH15] and are used to directly mesh the point cloud with a semi-regular triangular or quadrilateral mesh, without having to first convert it into an unstructured triangle mesh.

Volume The use of a field-aligned parametrization for remeshing purposes naturally extends to higher dimensions, where it has been applied for the creation of 3D hexahedral meshes [HTWB11, NRP11, LLX*12, JHW*14, KLF14]. The generation of suitable vector fields, however, becomes significantly more involved when moving to higher dimensions, as detailed in Section 12.

9.2. Deformation

In deformation applications, a *displacement field* describes the difference between two poses of a shape. Non-linear properties of a deformation, such as area/volume preservation, or isometry, are represented as properties of the displacement field. Divergence-free displacement fields do not change the volume to first-order. Hence, deformations that are approximately volume preserving can be generated by successively deforming a shape by small displacement fields, that are divergence free [vFTS06]. In a similar spirit, deformations which are as isometric as possible can be generated using approximate Killing vector fields [SBCBG11a]. Recently, [MERT14] created a generalized formulation that includes these two cases.

In reduced-order methods, the linear span of a set of deformation fields is used to generate low-dimensional subspaces of the space of all possible deformations. Restricting the system to be simulated within the subspace drastically reduces the dimension of the non-linear problems. Examples of subspace constructions include vibration modes and modal derivatives [BJ05, HSvTP11], subsampling [HSL*06, AWO*10, WWH*14] and linear blend skinning [KJ11, JBK*12]. In combination with a scheme for the efficient force evaluation, run times that are independent of the resolution of the mesh to be deformed can be achieved, which allows for real-time deformation of complex meshes.

Different force approximations schemes have been proposed, including the precomputation of the coefficients of reduced polynomials [BJ05], optimized cubature [AKJ08, vTSSH13], mesh coarsening [HSvTP11] and rotation clustering for the *as-rigid-as-possible* objective [JBK*12]. Recently, a method for real-time nonlinear shape interpolation was introduced [vTSSH15]. The proposed subspace construction involves the computation of the tangent vectors to the manifold of interpolating shapes.

9.3. Texture Mapping and Synthesis

Directional-field guided parametrization methods have been mainly applied for remeshing purposes (cf. Section 9.1). Nevertheless, the global, seamless parametrization that they produce can be used for other traditional graphics applications, such as texture mapping and mip-mapping. The regularity of the parametrization, and the simple seamlessness conditions, make it simple to define a seamless texture [RNLL10] that can be sampled at different resolutions without artefacts.

The stripe patterns occurring on many natural and biological objects, like plants (i.e. cactus ridges and maize), and animals (fish scale patterns, zebra stripes), can be synthesized with the guidance of a 2-direction field [RLL*06, KCPS15], describing smooth stripe directions. Fields with higher-order symmetries are useful for more general texture synthesis methods [WL01].

To synthesize more complex patterns, a shape grammar [SG71] can be applied to a surface using a vector field for guidance [LBZ*11]: the vector field locally defines a reference frame, that is used by the shape grammar to decide on the orientation of the pattern. This idea has been further generalized to synthesize volumetric textures composed of discrete elements [MWT11] (i.e. wood patterns, piles of stones).

9.4. Architectural Geometry

Polyhedral Meshes Directional fields are used extensively for architectural geometry purposes. One common application is remeshing with polyhedral meshes (meshes with flat polygonal faces). Planar faces are associated with the so-called *conjugate directions* [LPW*06], and thus an effort is invested to compute them on triangular meshes. Two vectors $\mathbf{u},\mathbf{v}$ are conjugate at a point p if $\mathrm{II}_p(\mathbf{u},\mathbf{v}) = 0$, where II is the second fundamental form [dC92]. In [ZSW10], two conjugate 2-direction fields are computed, only allowing for $\pm\frac{1}{2}$ singularities. In [LXW*11], The computation of the most general case of conjugate 2^2-direction fields is made possible. However, the optimization is nonlinear and nonsmooth (integer and binary variables were employed). In [DVPSH14], conjugate 2^2-vector fields are computed in a local-global manner, where smooth 2^2-drection fields are computed based on the complex polynomial representation (cf. Section 5.6) globally, and then projected to the closest conjugate directional fields locally. The aforementioned methods mostly targeted the design of planar quad meshes: Planar hexagonal meshes are considered in [LLW15, VBC15].

Self-Supporting Structures Special direction fields are used to establish surfaces that are in a stable equilibrium. In [VHWP12], principal directions of a relative surface operator serve as the conjugate field for planar quad meshing. In [PBSH13], 4-directional fields are computed for the meshing of the initial quad mesh, from which the structure would be built. The fields are computed as a balance between smoothness and adherence to prominent regions of the surface, that require the mesh to be built with a certain orientation. The problem is formulated in the language of DEC in [dGAOD13], where the new metric of the stress tensor is expressed by the hodge star. The metric is also altered by a 2^2-direction field to deform a mesh in a statics-aware way in [PTP*15].

Strip Surfaces Directional-field synthesis has also been employed to create architectural surfaces from strips corresponding to the surface curvature profile. In [PHD*10], a discrete *Jacobi field* is computed, and geodesic strips are extracted through it. A Jacobi field is a special vector field that indicates the density of geodesics at a point and in the direction it points to. Independently, they sharpen a 4-direction field to become a geodesic 2-direction field from which these strips are extracted.

9.5. Cultural Heritage

Parametrizations, which are guided by 4-direction fields, are used in cultural heritage to analyze and restore artifacts. The interactive system proposed in [PCCS11] allows a historian to sketch stripes on a 3D scanned model and use the stripes to guide the design of a 4-direction-field. This field is then used to flatten the stripe to a planar rectangle, where high-frequency details are easier to analyze.

A system to restore damaged historical parchments is presented in [PSP*14]. Each document is 3D scanned, and an optical character recognition (OCR) algorithm is used on the texture to extract a few recognizable characters. These characters provide clues on the deformation that the parchment underwent. This information is then propagated by interpolating a pair of 1-vector fields over the other parts of the document. A parametrization algorithm is then used to restore the document to its original flat geometry.

9.6. Other applications

Miscellaneous geometry-processing algorithms that use directional fields include surfacing [IBB15, PLS*15], mesh segmentation [SBCBG11b, ZZCJ14], parametrization [CSZ16], and shape analysis [HSvTP12]. Outside of geometry processing, directional fields have been used in procedural modeling [LBZ*11], crowd simulations [PvdBC*11], urban planning [YWVW13], digital fabrication [CPMS14], hair scanning [PCK*08], object design [FKS14], non-photorealistic rendering [HZ00, YCLJ12, Zha13, CYZL14], shading [MRMH12, RGB*14], and data analysis [FKSS13].

10. Visualization

Visualizing direction fields is a challenging problem that has a wide literature, especially in the scientific visualization community. In this Section, we give an overview of the most common rendering methods for symmetric fields on surfaces, and we refer an interested reader to [LDM*01, LKJ*05, LHZP07, FCL09, PL09, BCP*12, LV12] for an overview of 2D and 3D vector fields visualization methods.

10.1. Image-based Advection for Vector Fields

One of the most established techniques for visualizing vector fields is Line Integral Convolution (LIC) [CL93]. This technique generates a random image at the desired resolution. Then, for every pixel of the image, computes a streamline, i.e. a curve passing through the point whose tangent is aligned with the given vector field. The random image is then integrated over the streamline and the resulting value is associated to that pixel. The blur effect can be controlled by changing the length of the traced streamline. This technique might seem prohibitively expensive to apply to large images, but it is actually possible to efficiently map a variant of this method to modern graphics hardware by splitting the line integral to small steps that are performed in parallel over the entire domain [vW02].

This image-based approach can be directly extended to surfaces using a projection of the 3D surface in image-space: the streamlines can be then traced directly in the view plane, by using a projection of the vector field. Care has to be taken on the boundary of the domain, as detailed in [Wij03].

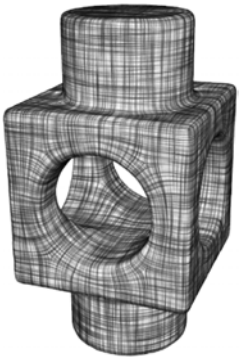
10.2. Image-based Advection for Tensor Fields

LIC is extended to general tensor fields [ZHT07] and N-RoSy fields [PZ11], by decomposing them into independent LIC-rendered vector fields, and then blending the resulting images. The blending should be optimized to increase the local contrast, as it decreases as more and more layers are blended. This techniques is particularly appealing for interactive applications, since it can render images in

real-time by implementing it in a GPU shader. For example, it is used in many interactive field-design algorithms [ZMT06, ZHT07, PZ11], where a field is interactively designed using radial basis functions, and the result is shown instantaneously.

10.3. Streamline Tracing

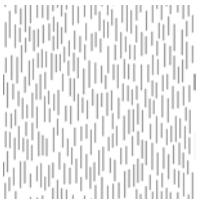
The previous techniques produce a dense visualization of the direction field by integrating a scalar field over the streamlines. However, a sparse set of separatrices can also be directly used to visualize the behavior of the field. This approach is preferred over LIC in many applications, such as non-photorealistic rendering using hatches [HZ00] or to natural phenomena synthesis [KCPS15]. The tracing can be done directly on the surface [RS14, MPZ14], or in image space [SLCZ09]. In both cases, it is important to enforce a uniform and not too dense sampling of the streamlines to avoid cluttering [MAD05, SLCZ09].



After tracing, streamlines can be replaced by a more complex geometry (such as a triangulated brush stroke with varying width) and then be rendered using a ray tracer. This method has been introduced in [CDS10], and then subsequently used in many other works, such as [KCPS13, KCPS15, JTPSH15].

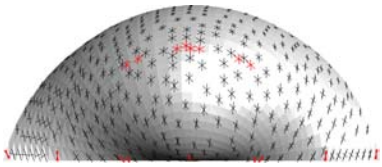
10.4. Texture-based Streamline Rendering

An alternative approach has been proposed in [PPTSH14], which replaces the streamline tracing with the computation of two parametrizations. These parametrizations are used to apply a stochastic texture map that is aligned with the direction field (see inset). The advantage of this approach is that it can naturally represent 2^2-direction fields, since it encodes skewness, scale, and anisotropy. Note that skewness can be represented directly with LIC, or any other tracing techniques, but scale and anisotropy cannot. As noted in [PZ11], these attributes might be mapped over color or line thickness, but the mapping would then be arbitrary and unintuitive.



The approach proposed in [PPTSH14] uses the spacing between the lines in one direction to indicate the length of the frame field component in the other direction. This cross-hatching tends to sketch rectangles, with the size and shape indicated by the frame field. A two-layered UV-mapping is propagated over the surface, following the frame field, i.e., the Jacobian of the parametrization of the vertices of a given triangle is close to the frame. In the final rendering, each fragment accesses the texture twice at the two UV locations, creating the two superimposed directions that form the cross-hatching.

10.5. Glyph Rendering



Glyphs, or collections of arrows, can be used to render the field at a subset of the points of a surface. While this approach is not as effective as the previous methods, due to the clutter introduced in the visualization, it is the only one that currently supports the visualization of non-symmetric (e.g. 1^3-) directional fields.

11. Algorithms & Comparisons

When some form of direction field synthesis is needed in an application, the question arises of which synthesis method (using which representation and which discretization) is suitable, and which is *best* suited. In the following we provide a desiderata-based guide to choosing the right method for various purposes, provide a property matrix (cf. Table 1), and empirically compare some of the properties of the state-of-the-art methods.

The first question that should be answered to find the best approach is whether the desired field topology is known in advance (cf. Section 11.2), or whether it needs to be optimized (cf. Section 11.3), i.e. automatically determined in a way conducive to the geometric objectives. This aspect has the most significant influence on the suitability of the various optimization strategies and field representations.

Since vector fields and direction fields are closely related, many algorithms for vector field synthesis can also be used to create direction fields, and direction field synthesis approaches can be employed to create (unit) vector fields. Certain properties, such as as-Killing-as-possible, however, fundamentally rely on magnitudal information and only apply to 1-vector fields. For the sake of clarity, we first consider algorithms specialized for 1-vector fields (cf. Section 11.1), and then discuss the more general problems of synthesizing directional fields (with all kinds of symmetries) in detail. We will also restrict our discussion to algorithms that can design fields on 2-manifolds or in the plane.

11.1. 1-Vector Fields

Fairness Fair tangent vector fields have been introduced in graphics for the purpose of decorating implicit surfaces in [Ped95]. The representation they chose is angle-based and supports directional constraints. Precise topological control is not supported. Field-guided texture synthesis on triangulated surface has been proposed [PFH00] (using RBF interpolation) and [Tur01] (using a diffusion equation). In both works, no control of the topology was possible, since the field is represented using extrinsic coordinates and then projected on the surface to make it tangential after the interpolation. A similar system focusing on fast, interactive manipulation is proposed in [ZMT06], where singularities can be manually introduced and moved over the surface. In [FSDH07] the sinks, sources and vortices of a vector field are controlled by prescribing its divergence and curl. To allow for additional control of the vector field, *e.g.* by prescribing directions, the divergence on curl are prescribed in a least squares sense. More recent approaches [AOCBC15] allow to compute the covariant derivative as an operator on vector fields, thus

Type	Method	Represent.	Topology	Constraints	Opt.
1-Vector	[PFH00]	Extrinsic	Partial	Dir	LS
1-Vector	[Ped95]	Extrinsic	None	None	LS
1-Vector	[Tur01]	Extrinsic	None	Dir	LS[1]
1-Vector	[FSDH07]	DEC	Partial	Dir	LS
1-Vector	[CML*07]	RBF	Partial	Dir	LS
1-Vector	[ZMT06]	RBF	Partial	Dir	LS
1-Vector	[ABCCO13]	Functional	Partial	Dir,Kil,Sym	LS
1-Vector	[CKW*12]	RBF	Partial	Dir	LS
1-Vector	[vFTS06]	Scalar	Partial	Dir,Div	LS
1-Vector	[WTD*06]	DEC	Partial	Dir,Harm[2]	LS
1-Vector	[BCBSG10]	DEC	None	Kil	Eigen
1-Vector	[SBCBG11a]	Extrinsic	None	Dir	LS
2-Direction	[PCK*08]	Struct. T.	None	Dir	LS
2-Vector	[YCLJ12]	Scalar	Partial	Dir	LS
N-Direction	[HZ00]	Angle	Partial	Dir	NL
N-Direction	[WL01]	Angle	Partial	Dir	NL
N-Direction	[RLL*06]	Complex	Partial	Dir	NL
N-Direction	[PZ07]	RBF	Partial	Dir	LS
N-Direction	[RVLL08]	Angle	Fixed	Dir	LS
N-Direction	[BZK09]	Angle	Full	Dir,Hom	MILP
N-Direction	[RVAL09]	Complex	Partial	Dir	NL
N-Direction	[LJX*10]	Complex	Fixed	Dir	NL
N-Direction	[CDS10]	DEC	Fixed	Dir	LS
N-Direction	[ZSW10]	DEC	Partial[3]	Dir	NL
N-Direction	[PZ11]	RBF	Partial	Dir	LS
N-Direction	[PLPZ12]	Angle	Full	Dir	MILP
N-Direction	[KCPS13]	Complex	None	Dir	Eigen
N-Direction	[MPP*13]	Angle	Full	Dir	MILP
N-Direction	[JTPSH15]	Angle	Full	Dir	MILP[4]
2^2-Vector	[PPTSH14]	Composite	Full	Dir	LS[5]
2^2-Vector	[IBB15]	Angle	Full	Dir,Part	MILP[6]
2^2-Vector	[JFH*15]	Composite	Full	Dir,Part	NL
2^2-Vector	[LXW*11]	Angle	Full	Dir,Conj	NLIP
2-Tensor	[dGLB*14]	DEC	None	Dir	LS
Tensor	[ZHT07]	RBF	Partial	Dir	LS
Directional	[DVPSH14]	Polynomial	Partial	Dir,Conj	LS[7]
Directional	[DVPSH15]	Polynomial	Partial	Dir,Part,Int	NL

Table 1: Summary of field synthesis algorithms. For each algorithm, from left to right: the type of field, the paper describing the algorithm, the representation used (cf. Section 5), the control over the singularity placement offered ("Partial": singularities can be prescribed, but additional ones can arise, "Full" means exact control is possible, "Fixed" means that the entire topology has to be specified), a list of supported constraint types (cf. Section 11, "Dir": directional constraints, "Part": partial directional constraints supported, "Conj": the resulting field is conjugate, "Int": the resulting field is integrable/curl-free, "Kil": the resulting field is as-Killing-as-possible, "Sym": the resulting field respects a given shape symmetry, "Hom": homology constrains can be given), the type of optimization used ("LS" = linear system, "NL" = non-linear, "MILP" = mixed-integer linear program,"NLIP" = non-linear integer program).

[1]) This algorighm uses a multi-resolution hierarchy.
[2]) The fields are harmonic only in the limit of increasing constraint weights.
[3]) The singularities are restricted to index $1/2$ and $-1/2$.
[4]) Scales linearly due to a custom MILP solver and a multi-grid hierarchy.
[5]) Linear in addition to the technique used to interpolate a N-direction field.
[6]) Two objectives are proposed, one is a MILP and the other is a NLIP.
[7]) Non-linear if the conjugacy constraint is used.

allowing to optimize for a Dirichlet type objective to synthesize smooth fields.

Note that many of the recent algorithms for general directional fields, as discussed in the remainder of this section, can be used for 1-vector fields as well. This can be of practical interest because some of them enable additional objectives and constraints.

Other Objectives Killing vector fields (and vector fields that are as-Killing-as-possible) naturally represent continuous intrinsic symmetries [BCBSG10] and can be used to generate close to isometric planar deformations [SBCBG11a]. The design of time-varying vector fields have been studied in [CKW*12]. Recently, a functional representation of vector fields has been proposed in [ABCCO13], which allows to naturally encode global constraints such as symmetry in an efficient way, at the price of losing precise local control over directional properties and the fields' topology.

11.2. Fixed Topology

11.2.1. Fairest N-Direction Field

A first algorithm to construct the fairest (in the sense of Section 8) N-symmetry field for a given field topology has been presented by Ray et al. [RVLL08]. It is based on the angle-based direction field representation (cf. Section 5.1), and only involves solving a sparse linear system with a size proportional to the size of the mesh, effectively minimizing the discrete Dirichlet energy. Note that without any further constraints the solution is not uniquely determined: any rotation by a globally constant angle applied to a field yields a field of equal smoothness. A directional constraint in a single face is sufficient to fix this one degree of freedom.

On a surface of genus $g > 0$ or with $b > 1$ boundary loops, there are $2g + b - 1$ topological degrees of freedom besides the singularity indices: the field's holonomy (or turning number) along the non-contractible cycles of a homology basis [RVLL08]. In some scenarios [CK14b] it is clear from the context how they shall be fixed, so this information is readily available. In other scenarios, like user-guided field design, requiring the manual fixing of these degrees of freedom can be unintuitive. Ray et al. [RVLL08] describe how they can be left free, i.e. only the singularities are prescribed, and in the end automatically be fixed in a reasonable way.

The direction field optimization method described by Bommes et al. [BZK09] is a generalization to the setting of completely free topology (neither singularity indices nor homology generator turning numbers are assumed as input). But, as the topology is explicitly represented in this method, it is possible to specify topological constraints. In the extreme the topology can be constrained entirely, effectively yielding a method equivalent to the algorithm of Ray et al.

Later Crane et al. [CDS10] presented a method that can be interpreted as a dual (or differential) formulation of the above algorithm: instead of solving for the per-face angles ϕ_i (cf. Section 5.1), it solves for the per-edge difference of incident face angles, the rotations δ_{ij}. Then choosing a direction ϕ_0 in a single face, all others are implied through these difference angles (cf. Section 5.1). "Triviality constraints" on the rotations must be taken into account to ensure global consistency, i.e. implication of a unique value ϕ_i per face.

These constraints fix the topology of the field (the relation between the rotations δ_{ij} and indices and turning numbers is treated in Section 6.2). The resulting field then is exactly the same as the one yielded by the above algorithm, i.e. again the algorithms are equivalent. This dual, connection-based point of view provides interesting insights, explicitly revealing why the singularities of cross fields (with topology optimized for smoothness) designate good irregular vertex configurations for quad meshes [Cam14, Chapter 4].

Optimization strategies for the smoothest field with fixed topology based on other forms of representation than the angle-based representation are harder to formulate. The main issue is measuring differences between vectors or directions across a specified number of periods, which in the angle-based setting amounts to a simple addition of multiples of 2π to the direct difference, but, e.g., with a Euclidean coordinate based representation is more complicated (Section 6.3). One noteworthy exception is the method described by Fisher et al. [FSDH07] using a 1-form representation (cf. Section 5.5). In this method the topology is prescribed using constraints on curl and divergence. It is, however, inherently limited to 1-direction/vector fields (N-symmetry fields for $N > 1$ are not supported).

Note that for fields with magnitudal component the optimization for smoothness without further constraints is not useful: the trivial solution (the zero-field) is a global optimum but obviously not of any interest.

11.2.2. Fairest N-Direction Field with Directional Constraints

Often one is interested in guiding the field synthesis in a certain way, influencing the directional behavior in a local or global manner. Of practical interest is the case of sparse hard constraints, i.e. fixing the field's direction in certain places, and the case of dense soft constraints, namely prescribing a (weighted) target field. Note that dense hard constraints obviously make no sense, while sparse soft constraints can be seen as a special case of the dense soft constraints scenario with low or zero weights in certain areas.

There is one significant obstacle to the consideration of such constraints in the fixed topology case: the topological configuration of the constraints needs to be known as well. For instance: how many turns does the field make between two directional constraints? Which of the N directions of an N-direction field is supposed to follow the prescribed direction? If this information is not available one expects the optimization algorithm to make an optimal (or at least good) choice. This implies a much harder optimization problem with discrete degrees of freedom [CK14b].

Dense Soft Constraints In order to take soft constraints into account, one simply adds a weighted data term to the Dirichlet term in the above angle-based algorithm [PLPZ12]. The optimization can then still be performed using a simple sparse linear system solve. If the constraint topology is unknown, a term $+\frac{k}{N}2\pi$ (effectively representing a free matching and period jump) needs to be added for each directional constraint, where k is an integer variable to be optimized per constraint [CK14b].

Sparse Hard Constraints Hard directional constraints can be taken into account by removing the corresponding angle variables from the optimization, fixing them to the desired values [BZK09, CDS10]. If the topological constellation is unknown, again a term $+\frac{k}{N}2\pi$ needs to be added per constraint.

11.2.3. Fairest 2^2-Direction Field with Directional Constraints

The algorithm for 2^2-direction field synthesis described in [IBB15] is based on the same explicit topology representation as [BZK09] which we considered in Section 11.2.1 for fairest N-direction fields. It is thus amenable to the same type of topological constraints, allowing for a complete prescription of the fields topology. It can therefore easily be used for the fixed topology use case. Hard and soft directional constraints can be taken into account in the same manner as well.

11.2.4. Fairest 2^2-Vector Field with Directional Constraints

Likewise, the algorithm for 2^2-vector field synthesis described in [PPTSH14] is based on the same explicit topology representation as [BZK09], allowing it to be used for the fixed topology use case.

11.3. Optimized Topology

For the optimization of a direction field's topology there are two fundamentally different approaches: the topology can be represented and optimized explicitly (i.e. there are variables in the optimization that directly express the topology), or it can be implicit. In the latter case one derives the field topology from the field geometry in a certain way. We discussed this in detail in Section 4.4.

The use of an explicit topology representation implies that a form of mixed optimization (with continuous and discrete degrees of freedom) is involved, because the topology of a direction field is not a continuous object (period jumps and matchings are discrete). This leads in general to optimization problems which are NP-hard. Nevertheless, it has been shown that quite efficient polynomial time approximations (based on a series of sparse linear system solves [BZK12]) can yield good results in practical applications, with an asymptotic behavior no different from algorithms with a purely continuous optimization, cf. Section 11.4.

An implicit topology representation is amenable to more efficient continuous optimization. However, depending on the representation and the constraints, the problem can become non-convex, and non-linear optimization, requiring a starting point, can become necessary. Of particular importance in this context is how easily objectives like parallelity can be expressed. As these objectives concern multiple (usually two) adjacent directionals, one needs to be able to compare these. In the context of fields with multiple directionals, to that end the matching (cf. Section 4.4) needs to be determined. The principal matching is inherent to some representations (cf. Sections 5.3, 5.2, and 5.6), and some form of fairness can be expressed in a linear manner with these representations, allowing for efficient optimization, even in a global manner [KCPS13].

11.3.1. Fairest N-Direction Field

The angle-based formulation of the Dirichlet energy (16) involves the period jumps (and matchings). When these are considered variables, the problem seeks to optimize field geometry *and* topology for smoothness. An approximative solution strategy for this setting was described by Bommes et al. [BZK09]. Exact optimization using,

e.g., Branch-and-Bound strategies is possible but rarely practical due to the large number of discrete variables.

Using one of the representations that inherently have a principal matching and rotation one can avoid this discrete optimization. However, in contrast to an angle based representation, these representations have an inherent magnitudal component that has to be dealt with. In particular, degeneration must be prevented, for instance using (non-linear) per-vector unity constraints [PZ07], a global unity constraint (which can be handled more efficiently) [KCPS13], or a sufficient number of constraints that explicitly fix the field's magnitude to non-zero values [DVPSH14] (but also affect its direction, reducing its fairness). Furthermore, the expressivity of this representation is limited: while the optimization with the above technique with explicit period jumps can yield and represent singularities of arbitrary index, only low-index singularities can arise here. The concrete limits depend on the valences of the mesh elements carrying the directionals. If the surface geometry or additional constraints require higher order singularities, clusters of low order singularities will arise instead in the respective region (cf. Section 6.3).

11.3.2. Fairest N-Direction Field with Directional Constraints

Directional constraints can be taken into account when using an angle-based representation, just as described for the fixed topology case in Section 11.2.2, with terms $+\frac{k}{N}2\pi$ per constraint.

Cartesian representations [PZ07, KCPS13, DVPSH14] include magnitude besides direction, such that value constraints fix not only the direction, but also the magnitude. When these vector constraints are then interpolated harmonically [DVPSH14], minimizing the Dirichlet energy (18), the prescribed magnitudes can bias the notion of fairness (cf. Figure 8) and in extreme cases lead to degeneracies (cf. Figure 6 (b)). The interesting question of whether hard directional constraints can be handled using an efficient Eigenvector problem formulation, akin to [KCPS13], is yet to be explored. Straightforward elimination of constrained variables from the system, as can be done in other formulations [BZK09], alters the structure of the system in a non-trivial way. The method of [PZ07] can handle hard directional constraints, but requires non-linear unity constraints.

11.3.3. Fairest N-Direction Field with Topological Constraints

Angle based methods can handle topological constraints due to the fact that the topology is completely described by the period jumps that are direct variables in the problem formulation. It is thus possible to prescribe the field's holonomy for arbitrary cycles on the surface, using a simple linear equality constraint on the sum of period jumps along the cycle [BZK09]. Most importantly, the indices of individual singularities can be fixed in this way (using the 1-ring cycles), and points and regions can be constrained to be regular (index 0).

For other representations where the topology is not explicitly accessible in the optimization, exerting such control is not easily possible and has in fact not been elaborated yet.

11.3.4. Fairest 2^2-Direction Field

For certain applications, it is useful to optimize for a 2^2-direction field instead of a 4-direction field. The method proposed in [IBB15] extends [BZK09] by adding an additional angle that represents the skewness of the field, i.e. how far it is from a 4-direction field. The formulation of the two papers is similar, and it gives full control over the topology, in addition to supporting hard and soft constraints, both on the directional alignment and on the skewness. Methods for the synthesis of 2^2-vector fields, discussed in the following section, could be used as well, by simply ignoring the magnitude of the resulting field.

11.3.5. Fairest 2^2-Vector Field

[DVPSH14] supports the design of 2^2-vector fields, and various field properties (direction, scale and skewness) can be controlled with soft or hard constraints. However, each constraint must fix all the properties, i.e. it is not possible to fix only the scale and not the skewness in a certain point. Topological constraints cannot be taken into account, due to the implicit topology representation. This is possible with methods which represent scale and skewness separately from an angle-based N-direction field with explicit period jumps [PPTSH14, JFH*15]. The method described by [ECBK14] is based on the same principle, but scale and skewness are optimized for a specific goal (scale-awareness) and not intended to be controlled or constrained by the user.

11.3.6. Fairest General Directional Field

A first representation and optimization method that supports general directional fields (without any symmetries) has recently been introduced [DVPSH14]. Direction and magnitude can be controlled in sparse and dense, hard and soft manners. Topological control is not possible.

11.4. Comparative Analysis

In this subsection, we experimentally compare field design algorithms with respect to scalability, quality of the results and singularity placement to provide additional insights on the similarities and differences between the algorithms.

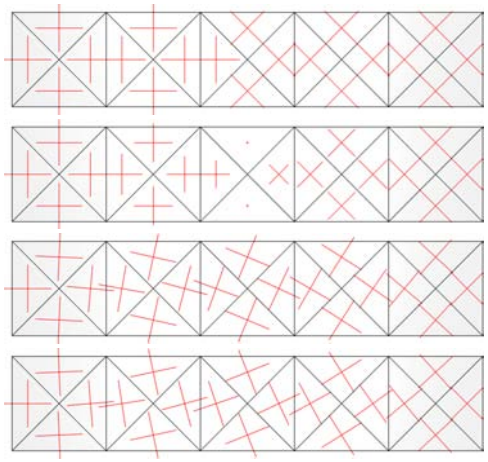


Figure 6: Interpolation of a 4-direction field on a planar triangle strip, with the leftmost and rightmost faces constrained.

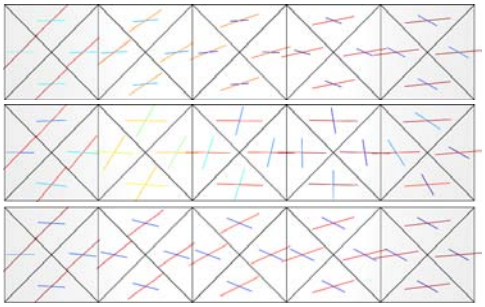


Figure 7: Interpolation of a 2^2-vector field on a planar triangle strip, with the available methods that support this field type. The leftmost and rightmost faces are constrained. The vectors are colored according to their magnitude, to illustrate differences in scale.

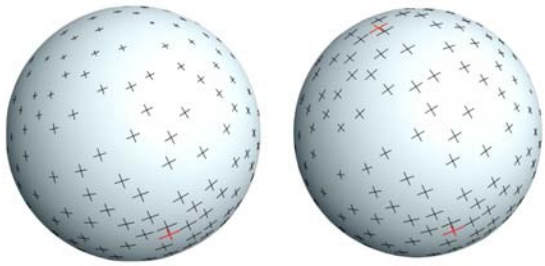


Figure 8: 2^2-vector fields synthesized with [DVPSH14]: the red crosses are the constraints. If the constraints are very sparse (left), we observe a reduction in scale caused by the fairness objective. While this is not a major practical problem (this phenomenon disappears with just a few more constraints), it would be preferable to have a fairness objective that preserves scale. Image courtesy of [DVPSH14].

Constrained 4-Direction Fields In Figure 6, we compare different methods for designing a 4-direction field on a planar strip. The leftmost and rightmost triangles are fixed, and the interior part is designed by the algorithms. For fair comparison, we use a face-based variant of [KCPS13], and employ hard vector constraints on a sparse set of faces, in a way equivalent to using [DVPSH14] in the subspace of N-RoSy fields. Note that (a) is thus identical to (b) in this specific case, up to a normalization of the vectors. It is interesting to note the different behavior between complex representations (a), (b) and angle representations (c),(d): the latter naturally interpolates the rotation in the constraints, while the former mix scale changes in the interpolation. Interestingly, the scale in (b) rapidly drops in the middle: this is an undesirable feature in many applications which can be ameliorated by adding additional constraints (Figure 8). The normalization in (a) partially conceals the problem, but produces highly unfair fields, with triangle to triangle differences of up to 45 degrees.

Constrained 2^2-Vector Fields We perform the same experiment for 2^2-Vector fields (Figure 7) and obtain a similar behavior: the angle based representation (c) favors interpolating a rotation, while complex based representations (a), (b) favor the interpolation of skewness and scale.

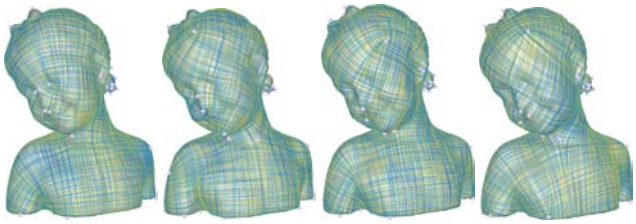
Singularities The distribution and number of singularities of a directional field is one of the main criteria for evaluating its quality for many applications. We experimented with different methods (an example for 4-direction fields is shown in Figure 9) and observed that angle based approaches (a), (b) tend to introduce less singularities than complex-number representations (c), (d). However, the greedy solution strategies that are used for angle based representations can get stuck in local minima, while the convex formulation of [KCPS13] guarantees that the optimal solution (with respect to a specific fairness objective) can be found. This is particularly noticeable in symmetric shapes (Figure 10), where the non-optimal solutions might have non symmetric singularity placement.

Integrability When a direction field is used to create a field-aligned parametrization, its integrability, i.e. how far it is from being a gradient of a set of scalar functions, plays an important role. While specialized methods can produce integrable fields [DVPSH15], they require an expensive optimization. In the literature, many other methods have been used to design fields to guide parametrizations, and we compare a few in Figure 11, where we plot and measure the Poisson error obtained when using the fields as target gradients. This error is measured as the average L^2-norm between the field and the gradient of the scalar function retrieved after integration.

The complex representation (a), (b) tends to produce results that are more integrable than angle based. In particular, measuring the smoothness using the complex representation and then avoiding normalization (b), consistently produces results that are more integrable: this suggest that the fairness measure in the complex representation indirectly favors integrability but this property is partially lost in the normalization. For a more extensive comparison of the two representations for the purpose of isotropic quadrangulation, we refer to the experimental section of [JTPSH15].

Scalability We compare the efficiency of different field design algorithms in a controlled experiment (Figure 12), where all algorithms are executed on three series of meshes obtained decimating three high-resolution meshes.

The multi-resolution hierarchy, combined with the embarrassingly parallel nature of the optimization, makes the algorithm proposed in [JTPSH15] noticeably faster than the others, both in absolute running time and in term of asymptotic running time. In-



(a) [BZK09] 48 singularities | (b) [JTPSH15] 42 singularities | (c) [KCPS13] 62 singularities | (d) [DVPSH15] 56 singularities

Figure 9: Singularities of a 4-direction field. The fields have been constrained on the same 14 faces (uniformly sampled) for all methods, where the direction is constrained to be the projection of an horizontal vector.

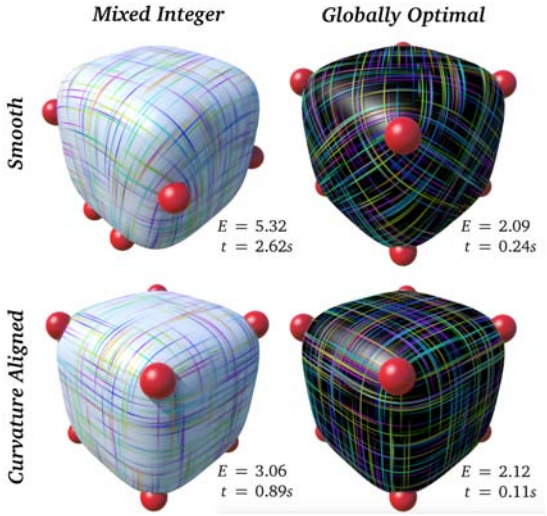


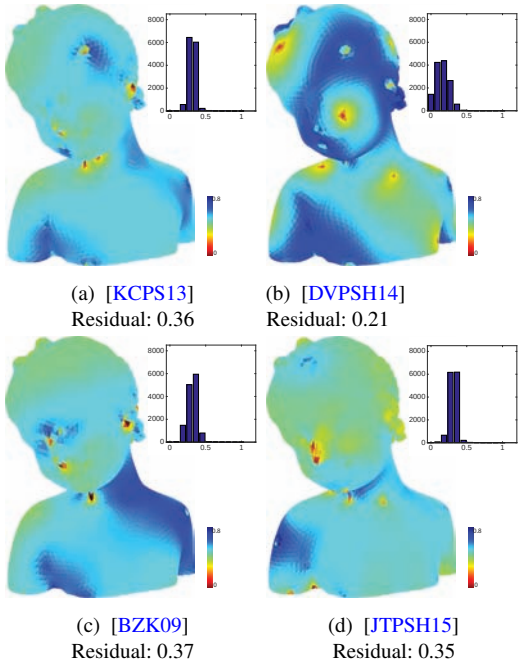
Figure 10: The global optimality of [KCPS13] (with implicit topology) ensures that symmetric singularity placement is obtained on a symmetric shape, while approaches like [BZK09] (with explicit topology representation based on period jumps) might fail to find the optimal solution due to the involved discrete optimization. Image courtesy of [KCPS13].

terestingly, all other algorithms have similar asymptotic behavior, suggesting that for these field design problems the performance advantage of methods that requires to solve a linear system [RLL*06, CDS10, KCPS13, DVPSH14] compared with the ones that require a mixed-integer solver [BZK09, JTPSH15] is only a constant multiplicative factor that does not depend on the resolution of the dataset. [DVPSH15] is considerably slower than all other methods due to the non-linear optimization used to enforce integrability.

12. Open Questions

We have presented the state-of-the-art in directional field synthesis, design, and processing. In this section, we discuss possible generalizations of existing methods, and interesting unsolved problems.

Topology Control Currently, there is no representation that incorporates the advantages of the Cartesian representation methods (namely, representation of non-unit vector fields, and having convex and continuous objective functions), while providing direct control over the field topology. This problem is exacerbated by the fact that methods that advocate fairness optimization as a tool for automatic singularity placement, empirically tend to introduce many low-degree singularities, possibly because of the sampling problem discussed in Section 6.3. Mixed-integer angle-based methods are highly non-linear, and linear angle-based methods require the manual prescription of singularities. This prescription requires some expertise, and is not directly associated with the fairness of the vector field. Hence, there is a demand for a representation which is both general enough to include non-unit vector fields, is equipped with an efficient fairness objective, and that allows for more control over the topology.



(a) [KCPS13] Residual: 0.36
(b) [DVPSH14] Residual: 0.21
(c) [BZK09] Residual: 0.37
(d) [JTPSH15] Residual: 0.35

Figure 11: Integration error for the fields shown in Fig. 9: the color shows how much the field deviates from being the gradient of a scalar field. We omitted [DVPSH15] from this comparison since it directly optimizes for integrability.

A promising direction is the exploration of fairness objectives which are not as-parallel-as-possible, but involve some notion of influence on singularities. The conformal energy in [KCPS13], as well as objectives with alternative notions of parallelism [RVAL09, ECBK14] show this effect to some extent.

Sampling and Convergence Period jumps, principal matching, and discrete definitions of singularities all bridge the gap between

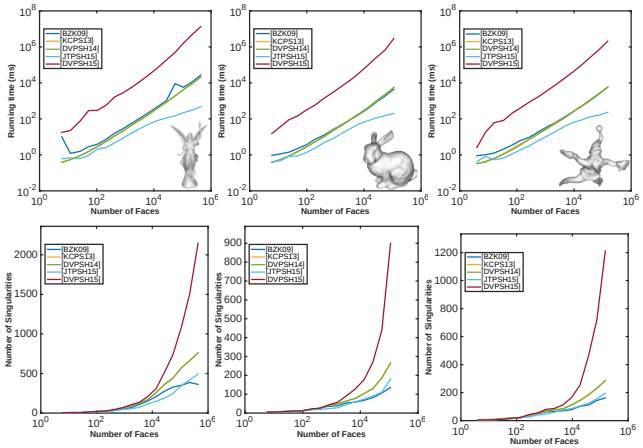


Figure 12: Different field design methods on three series of meshes with increasing resolution. Top: log-log plot of the running times, middle: mumber of singularities

the discrete and the continuous. However, we have gathered the inconsistencies that arise from these sampling methods. What is lacking is a consistent sampling theorem for 1^N-directional fields on discrete meshes that would answer the classical question from discrete signal processing: given a continuous field, what is a proper sampling, in the sense that it allows for a full reconstruction of the original field? Evidently, the answer to these question is correlated with general sampling problems on discrete meshes. The insights on sampling we presented in Section 6.3 could be a background to set some ground rules for a future theory on field sampling and reconstruction.

Geometry of Directional Fields There is a multitude of open questions related to the discretization of various differential operators on directional fields. First, there are operators, which map from vector fields to vector fields, that still do not have a satisfactory discretization on surfaces. One example is the Lie bracket of two vector fields, which measures the commutativity of the flow of the vector fields, and is important for parametrization. Discretizations for the Lie bracket were suggested in [ABCCO13, dGLB*14, AOCBC15]. However, the first and last compute an operator on functions, while the second requires the solution of a sparse linear system, and therefore yields an operator which is not local. A systematic study of the geometric structure of N-vector fields in its different forms (described in Section 2), and the properties of operators acting on them, is still missing. The recent preprint of Knöppel and Pinkall [KP15] provides a classification of discrete complex line bundles over simplicial complexes.

Suitability for Parametrization and Meshing The application area that perhaps benefited the most from recent advances in directional field synthesis is field-guided parametrization; in particular, for purposes of quadrilateral mesh or layout generation. In this context, not only the directional information of the field is exploited: its topology, i.e. its singularities and holonomy, is used to define a suitable parametrization domain, and to decide over number, type, and position of irregular vertices in high quality semi-regular quad meshes. It is commonly assumed that the topology of an arbitrarily synthesized directional field is suitable for that purpose. However, the topological structure of a seamless parametrization or a quad mesh is slightly more restricted. For instance, not every singularity configuration is valid in this context [JT73, IKR*12], and global holonomy can likewise be an issue [KNP07, MPZ14]. Precise conditions still need to be discerned in detail, before specialized synthesis methods can be developed. For the time being, only post-processing adjustments can be made to remedy problematic situations, for instance, by introducing additional singularities [MPZ14].

3D Generalization The generalization of the discussed concepts to the three-dimensional setting, for instance based on tetrahedral meshes, comes with a number of severe complications.

- The angle-based representation (cf. Section 5.1) that allows for a field definition without ambiguities regarding rotational periods, does not extend to 3D. A similar problem arises for the angle-valued period jumps. This precludes any form of precise topological control in 3D directional field synthesis; best-effort approaches are available [WTHS04], but guarantees cannot be given.
- Directions in 2D can be parameterized by an angle, and vectors by Cartesian coordinates or complex numbers. However, representations for 3D orientations, such as matrices, quaternions [KLF14], tensors [PCK*08], or spherical harmonics coefficients [HTWB11], require additional constraints (orthogonal and unit determinant, unit length, or rotation-equivalent, respectively). Such constraints lead to more complicated and less efficient (non-linear or non-convex) optimization procedures. For instance, interleaving steps for objective reduction and re-projection onto the constraint manifold.
- There are some pathological cases of field topologies which do not lead to valid parametrizations and quad mesh generation. When moving from 2D to 3D this small gap turns into a very large gap: in fact, only a small subset of possible 3D directional field topologies is suitable for these purposes, as discussed in [NRP11] and tackled partially in [LLX*12, JHW*14]. Reliable solutions are yet to be found.

13. Available Implementations

There are freely available implementations for a number of the synthesis and visualization methods discussed in this report. We have collected them, and list them in the following table. The name of each paper is a hyperlink to the webpage containing the respective source code.

1-Vector Fields
As-Killing-As-Possible Vector Fields for Planar Deformation [SBCBG11a]
Design of 2D Time-Varying Vector Fields [CKW*12]
An Operator Approach to Tangent Vector Field Processing [ABCCO13]
***N*-Directional Fields**
Rotational Symmetry Field Design on Surfaces [PZ07]
Mixed-Integer Quadrangulation (+Solver) [BZK09]
Interactive Visualization of Rotational Symmetry Fields on Surfaces [PZ11]
Trivial Connections on Discrete Surfaces [CDS10]
Globally Optimal Direction Fields [KCPS13]
Instant Field-Aligned Meshes [JTPSH15]
2^2-Directional Fields
Frame Fields: Anisotropic and Non-Orthogonal Cross Fields [PPTSH14]
Regularized Curvature Fields from Rough Concept Sketches [IBB15]
1^N-Vector Fields
Designing *N*-PolyVector Fields with Complex Polynomials [DVPSH14]
Integrable PolyVector Fields [DVPSH15]

Acknowledgements

The authors would like to thank Olga Sorkine-Hornung and Keenan Crane for their insightful comments, Marco Tarini, Nico Pietroni, Wenzel Jakob and Kevin Wallimann for contributing their implementations. This work was supported in part by the ERC Starting Grant iModel (StG-2012-306877), the German Research Foundation (DFG grant GSC 111 Aachen Institute for Advanced Study in Computational Engineering Science), ISF grant 699/12, and Marie Curie CIG 303511.

References

[ABCCO13] AZENCOT O., BEN-CHEN M., CHAZAL F., OVSJANIKOV M.: An operator approach to tangent vector field processing. *Computer Graphics Forum 32*, 5 (2013). 11, 13, 14, 15, 19, 24

[ACSD*03] ALLIEZ P., COHEN-STEINER D., DEVILLERS O., LÉVY B., DESBRUN M.: Anisotropic polygonal remeshing. *ACM Transactions on Graphics 22*, 3 (2003). 16

[AF02] AGRICOLA I., FRIEDRICH T.: *Global Analysis: Differential Forms in Analysis, Geometry, and Physics*. AMS, 2002. 4, 5

[AFW06] ARNOLD D. N., FALK R. S., WINTHER R.: Finite element exterior calculus, homological techniques, and applications. *Acta numerica 15*, 1 (2006). 10

[AKJ08] AN S. S., KIM T., JAMES D. L.: Optimizing cubature for efficient integration of subspace deformations. *ACM Transactions on Graphics 27*, 5 (2008), 1–10. 16

[AOCBC15] AZENCOT O., OVSJANIKOV M., CHAZAL F., BEN-CHEN M.: Discrete derivatives of vector fields on surfaces – an operator approach. *ACM Transactions on Graphics 34*, 3 (May 2015). 13, 18, 24

[AWO*10] ADAMS B., WICKE M., OVSJANIKOV M., WAND M., SEIDEL H., GUIBAS L. J.: Meshless shape and motion design for multiple deformable objects. *Computer Graphics Forum 29*, 1 (2010), 43–59. 16

[AWO*14] AZENCOT O., WEISSMANN S., OVSJANIKOV M., WARDETZKY M., BEN-CHEN M.: Functional fluids on surfaces. *Computer Graphics Forum 33*, 5 (2014). 11, 15

[BCBSG10] BEN-CHEN M., BUTSCHER A., SOLOMON J., GUIBAS L.: On discrete killing vector fields and patterns on surfaces. *Computer Graphics Forum 29*, 5 (2010). 10, 13, 14, 15, 19

[BCE*13] BOMMES D., CAMPEN M., EBKE H.-C., ALLIEZ P., KOBBELT L.: Integer-grid maps for reliable quad meshing. *ACM Transactions on Graphics 32*, 4 (2013). 16

[BCP*12] BRAMBILLA A., CARNECKY R., PEIKERT R., VIOLA I., HAUSER H.: Illustrative Flow Visualization: State of the Art, Trends and Challenges. In *Eurographics 2012 – State of the Art Reports* (2012). 1, 17

[BJ05] BARBIČ J., JAMES D. L.: Real-time subspace integration for St. Venant-Kirchhoff deformable models. *ACM Transactions on Graphics 24*, 3 (2005), 982–990. 16

[BK01] BOTSCH M., KOBBELT L.: Resampling feature regions in polygonal meshes for surface anti-aliasing. *Computer Graphics Forum 20*, 3 (2001). 16

[BLP*13] BOMMES D., LÉVY B., PIETRONI N., PUPPO E., SILVA C., TARINI M., ZORIN D.: Quad-mesh generation and processing: A survey. *Computer Graphics Forum 32*, 6 (2013). 3

[BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. *ACM Transactions on Graphics 28*, 3 (2009). 9, 10, 14, 15, 16, 19, 20, 21, 22, 23, 24

[BZK12] BOMMES D., ZIMMER H., KOBBELT L.: Practical mixed-integer optimization for geometry processing. In *Proceedings of the 7th international conference on Curves and Surfaces* (Berlin, Heidelberg, 2012), Springer-Verlag, pp. 193–206. 20

[Cam14] CAMPEN M.: *Quad Layouts: Generation and Optimization of Conforming Quadrilateral Surface Partitions*. Shaker Verlag, 2014. 20

[CBK12] CAMPEN M., BOMMES D., KOBBELT L.: Dual loops meshing: quality quad layouts on manifolds. *ACM Transactions on Graphics 31*, 4 (2012). 16

[CBK15] CAMPEN M., BOMMES D., KOBBELT L.: Quantized global parametrization. *ACM Transactions on Graphics 34*, 6 (2015). 16

[CdGDS13] CRANE K., DE GOES F., DESBRUN M., SCHRÖDER P.: Digital geometry processing with discrete exterior calculus. In *ACM SIGGRAPH 2013 Courses* (2013), SIGGRAPH '13. 12

[CDS10] CRANE K., DESBRUN M., SCHRÖDER P.: Trivial connections on discrete surfaces. *Computer Graphics Forum 29*, 5 (2010). 8, 9, 12, 14, 15, 18, 19, 20, 23, 24

[CHK13] CAMPEN M., HEISTERMANN M., KOBBELT L.: Practical anisotropic geodesy. *Computer Graphics Forum 32*, 5 (2013). 16

[CK14a] CAMPEN M., KOBBELT L.: Interactive design of quad layouts using elastica strips. *ACM Transactions on Graphics 33*, 6 (2014). 16

[CK14b] CAMPEN M., KOBBELT L.: Quad layout embedding via aligned parameterization. *Computer Graphics Forum 33*, 8 (2014). 16, 19, 20

[CKW*12] CHEN G., KWATRA V., WEI L., HANSEN C. D., ZHANG E.: Design of 2d time-varying vector fields. *IEEE Transactions on Visualization and Computer Graphics 18*, 10 (2012). 19, 24

[CL93] CABRAL B., LEEDOM L. C.: Imaging vector fields using line integral convolution. In *Proc. SIGGRAPH '93* (1993). 17

[CML*07] CHEN G., MISCHAIKOW K., LARAMEE R. S., PILARCZYK P., ZHANG E.: Vector field editing and periodic orbit extraction using morse decomposition. *IEEE Transactions on Visualization and Computer Graphics 13*, 4 (2007). 19

[COC15] CORMAN E., OVSJANIKOV M., CHAMBOLLE A.: Continuous Matching via Vector Field Flow. *Computer Graphics Forum* (2015). 11, 15

[CPMS14] CIGNONI P., PIETRONI N., MALOMO L., SCOPIGNO R.: Field-aligned mesh joinery. *ACM Transactions on Graphics 33*, 1 (2014). 17

[CSAD04] COHEN-STEINER D., ALLIEZ P., DESBRUN M.: Variational shape approximation. *ACM Transactions on Graphics 23*, 3 (2004). 16

[CSZ16] CAMPEN M., SILVA C. T., ZORIN D.: Bijective maps from simplicial foliations. *ACM Transactions on Graphics, to appear 35*, 4 (2016). 7, 17

[CYZL14] CHI M., YAO C., ZHANG E., LEE T.: Optical illusion shape texturing using repeated asymmetric patterns. *The Visual Computer 30*, 6-8 (2014). 17

[D'A00] D'AZEVEDO E. F.: Are bilinear quadrilaterals better than linear triangles? *J. Sci. Comput. 22*, 1 (2000). 16

[dC92] DO CARMO M. P.: *Riemannian Geometry*. Birkhäuser, 1992. 3, 4, 13, 17

[dGAOD13] DE GOES F., ALLIEZ P., OWHADI H., DESBRUN M.: On the equilibrium of simplicial masonry structures. *ACM Transactions on Graphics 32*, 4 (2013). 17

[dGDT15] DE GOES F., DESBRUN M., TONG: Vector field processing on triangle meshes. *SIGGRAPH Asia 2015 Course Notes* (2015). 2, 10

[dGLB*14] DE GOES F., LIU B., BUDNINSKIY M., TONG Y., DESBRUN M.: Discrete 2-tensor fields on triangulations. *Computer Graphics Forum 33*, 5 (2014). 10, 13, 19, 24

[DHLM05] DESBRUN M., HIRANI A., LEOK M., MARSDEN J.: Discrete exterior calculus. preprint, arXiv:math.DG/0508341, 2005. 7, 12

[DVPSH14] DIAMANTI O., VAXMAN A., PANOZZO D., SORKINE-HORNUNG O.: Designing *N*-PolyVector fields with complex polynomials. *Computer Graphics Forum 33*, 5 (2014). 3, 5, 11, 12, 14, 15, 16, 17, 19, 21, 22, 23, 24

[DVPSH15] DIAMANTI O., VAXMAN A., PANOZZO D., SORKINE-HORNUNG O.: Integrable PolyVector fields. *ACM Transactions on Graphics 34*, 4 (2015). 15, 16, 19, 22, 23, 24

[ECBK14] EBKE H.-C., CAMPEN M., BOMMES D., KOBBELT L.: Level-of-detail quad meshing. *ACM Transactions on Graphics 33*, 6 (2014). 14, 16, 21, 23

[FCL09] FORSBERG A., CHEN J., LAIDLAW D.: Comparing 3d vector field visualization methods: A user study. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009). 17

[FH99] FARIN G., HANSFORD D.: Discrete coons patches. *Computer Aided Geometric Design 16*, 7 (1999), 691–700. 15

[FKS14] FU L., KARA L. B., SHIMADA K.: Modeling flow features with user-guided streamline parameterization. *Computer-Aided Design 46* (2014). 17

[FKSS13] FERREIRA N., KLOSOWSKI J. T., SCHEIDEGGER C. E., SILVA C. T.: Vector field *k*-means: Clustering trajectories by fitting multiple vector fields. *Computer Graphics Forum 32*, 3 (2013), 201–210. 17

[FSDH07] FISHER M., SCHRÖDER P., DESBRUN M., HOPPE H.: Design of tangent vector fields. *ACM Transactions on Graphics 26*, 3 (2007). 10, 12, 14, 15, 18, 19, 20

[Ful95] FULTON W.: *Algebraic Topology: A First Course*. Springer, 1995. 5

[GE88] GIANNAKOPOULOS A. E., ENGEL A. J.: Directional control in grid generation. *J. Comput. Phys. 74*, 2 (1988). 16

[GGT06] GORTLER S. J., GOTSMAN C., THURSTON D.: Discrete one-forms on meshes and applications to 3d mesh parameterization. *Computer Aided Geometric Design 23*, 2 (2006). 10

[GH97] GARLAND M., HECKBERT P. S.: Surface simplification using quadric error metrics. In *Proceedings of SIGGRAPH* (1997), pp. 209–216. 7

[GK95] GRANLUND G. H., KNUTSSON H.: *Signal Processing for Computer Vision*. Kluwer Academic Publishers, Norwell, MA, USA, 1995. 10

[GRK12] GRUSHKO C., RAVIV D., KIMMEL R.: Intrinsic local symmetries: A computational framework. In *Proceedings of the 5th Eurographics Conference on 3D Object Retrieval* (2012), EG 3DOR'12, Eurographics Association. 13

[GY02] GU X., YAU S.-T.: Computing conformal structure of surfaces. *Communications in Information and Systems 2*, 2 (2002), 121–146. 10

[Hir03] HIRANI A. N.: *Discrete exterior calculus*. PhD thesis, California Institute of Technology, 2003. 10, 12

[HSL*06] HUANG J., SHI X., LIU X., ZHOU K., WEI L.-Y., TENG S.-H., BAO H., GUO B., SHUM H.-Y.: Subspace gradient domain mesh deformation. *ACM Transactions on Graphics 25*, 3 (2006). 16

[HSvTP11] HILDEBRANDT K., SCHULZ C., VON TYCOWICZ C., POLTHIER K.: Interactive surface modeling using modal analysis. *ACM Transactions on Graphics 30*, 5 (2011), 119:1–119:11. 16

[HSvTP12] HILDEBRANDT K., SCHULZ C., VON TYCOWICZ C., POLTHIER K.: Modal shape analysis beyond Laplacian. *Computer Aided Geometric Design 29*, 5 (2012), 204–218. 17

[HTWB11] HUANG J., TONG Y., WEI H., BAO H.: Boundary aligned smooth 3d cross-frame field. *ACM Transactions on Graphics 30*, 6 (2011). 11, 16, 24

[HZ00] HERTZMANN A., ZORIN D.: Illustrating smooth surfaces. In *Proc. SIGGRAPH 2000* (2000). 3, 9, 17, 18, 19

[IBB15] IARUSSI E., BOMMES D., BOUSSEAU A.: Bendfields: Regularized curvature fields from rough concept sketches. *ACM Transactions on Graphics 34*, 3 (2015). 9, 14, 15, 17, 19, 20, 21, 24

[IKR*12] IZMESTIEV I., KUSNER R. B., ROTE G., SPRINGBORN B., SULLIVAN J. M.: There is no triangulation of the torus with vertex degrees 5, 6, ..., 6, 7 and related results: Geometric proofs for combinatorial theorems. *ArXiv e-prints* (2012). 24

[Jae13] JAENICH K.: *Vector Analysis*. Springer, 2013. 5

[JBK*12] JACOBSON A., BARAN I., KAVAN L., POPOVIĆ J., SORKINE O.: Fast automatic skinning transformations. *ACM Transactions on Graphics 31*, 4 (2012), 77:1–77:10. 16

[JFH*15] JIANG T., FANG X., HUANG J., BAO H., TONG Y., DESBRUN M.: Frame field generation through metric customization. *ACM Transactions on Graphics 34*, 4 (2015). 10, 14, 16, 19, 21

[JHW*14] JIANG T., HUANG J., WANG Y., TONG Y., BAO H.: Frame field singularity correctionfor automatic hexahedralization. *IEEE Transactions on Visualization and Computer Graphics 20*, 8 (2014). 16, 24

[Jos08] JOST J.: *Riemannian geometry and geometric analysis*. Springer, 2008. 3, 4

[JT73] JUCOVIČ E., TRENKLER M.: A theorem on the structure of cell-decompositions of orientable 2-manifolds. *Mathematika 20* (1973). 24

[JTPSH15] JAKOB W., TARINI M., PANOZZO D., SORKINE-HORNUNG O.: Instant field-aligned meshes. *ACM Transactions on Graphics 34*, 6 (Nov. 2015). 15, 16, 18, 19, 21, 22, 23, 24

[KCPS13] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Globally optimal direction fields. *ACM Transactions on Graphics 32*, 4 (2013). 7, 8, 9, 11, 12, 14, 15, 18, 19, 20, 21, 22, 23, 24

[KCPS15] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Stripe patterns on surfaces. *ACM Transactions on Graphics 34* (2015). 16, 18

[KJ11] KIM T., JAMES D. L.: Physics-based character skinning using multi-domain subspace deformations. In *Symposium on Computer Animation* (2011), pp. 63–72. 16

[KLF13] KOWALSKI N., LEDOUX F., FREY P.: A PDE based approach to multidomain partitioning and quadrilateral meshing. In *Proceedings of the 21st International Meshing Roundtable* (2013). 9

[KLF14] KOWALSKI N., LEDOUX F., FREY P.: Block-structured hexahedral meshes for CAD models using 3d frame fields. *Procedia Engineering 82* (2014). 23rd International Meshing Roundtable (IMR23). 16, 24

[KNP07] KÄLBERER F., NIESER M., POLTHIER K.: Quadcover - surface parameterization using branched coverings. *Computer Graphics Forum 26*, 3 (2007). 16, 24

[Knu95] KNUPP P.: Mesh generation using vector fields. *J. Comput. Phys. 119*, 1 (1995). 16

[KP15] KNÖPPEL F., PINKALL U.: Complex line bundles over simplicial complexes and their applications. preprint, arXiv:1506.07853, 2015. 7, 24

[Kue05] KUEHNEL W.: *Differential Geometry: Curves - Surfaces - Manifolds*. AMS, 2005. 3

[LBZ*11] LI Y., BAO F., ZHANG E., KOBAYASHI Y., WONKA P.: Geometry synthesis on surfaces using field-guided shape grammars. *IEEE Transactions on Visualization and Computer Graphics 17*, 2 (2011). 17

[LDM*01] LAIDLAW D. H., DAVIDSON J. S., MILLER T. S., DA SILVA M., KIRBY R. M., WARREN W. H., TARR M.: Quantitative comparative evaluation of 2d vector field visualization methods. In *Proceedings of the Conference on Visualization '01* (2001), VIS '01, IEEE Computer Society. 17

[LHZP07] LARAMEE R. S., HAUSER H., ZHAO L., POST F. H.: Topology-based flow visualization, the state of the art. In *Topology-based Methods in Visualization*, Mathematics and Visualization. 2007. 1, 17

[LJX*10] LAI Y.-K., JIN M., XIE X., HE Y., PALACIOS J., ZHANG E., HU S.-M., GU X.: Metric-driven rosy field design and remeshing. *IEEE Transactions on Visualization and Computer Graphics 16*, 1 (2010). 19

[LKJ*05] LAIDLAW D., KIRBY R., JACKSON C., DAVIDSON J., MILLER T., DA SILVA M., WARREN W., TARR M.: Comparing 2d vector field visualization methods: a user study. *IEEE Transactions on Visualization and Computer Graphics 11*, 1 (Jan 2005). 17

[LLW15] LI Y., LIU Y., WANG W.: Planar hexagonal meshing for architecture. *IEEE Transactions on Visualization and Computer Graphics 21*, 1 (Jan 2015). 16, 17

[LLX*12] LI Y., LIU Y., XU W., WANG W., GUO B.: All-hex meshing using singularity-restricted field. *ACM Transactions on Graphics 31*, 6 (2012). 16, 24

[LLZ*11] LI E., LÉVY B., ZHANG X., CHE W., DONG W., PAUL J.-C.: Meshless quadrangulation by global parameterization. *Computers & Graphics* (2011). 16

[LPW*06] LIU Y., POTTMANN H., WALLNER J., YANG Y.-L., WANG W.: Geometric modeling with conical meshes and developable surfaces. *ACM Transactions on Graphics 25*, 3 (2006). 17

[LRL06] LI W.-C., RAY N., LÉVY B.: Automatic and interactive mesh to t-spline conversion. In *Proc. SGP '06* (2006). 16

[LV12] LAIDLAW D. H., VILANOVA A.: *New developments in the visualization and processing of tensor fields*. Springer Science & Business Media, 2012. 10, 17

[LVRL06] LI W. C., VALLET B., RAY N., LÉVY B.: Representing Higher-Order Singularities in Vector Fields on Piecewise Linear Surfaces. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006). 8, 9, 12

[LXW*11] LIU Y., XU W., WANG J., ZHU L., GUO B., CHEN F., WANG G.: General planar quadrilateral mesh design using conjugate direction field. *ACM Transactions on Graphics 30*, 6 (2011). 3, 9, 14, 16, 17, 19

[MAD05] MEBARKI A., ALLIEZ P., DEVILLERS O.: Farthest point seeding for efficient placement of streamlines. In *Visualization, 2005. VIS 05. IEEE* (Oct 2005). 18

[Max99] MAX N.: Weights for computing vertex normals from facet normals. *J. Graphics, GPU, & Game Tools 4*, 2 (1999), 1–6. 7

[Mer01] MERCAT C.: Discrete Riemann surfaces and the Ising model. *Communications in Mathematical Physics 218*, 1 (2001), 177–216. 10

[MERT14] MARTINEZ ESTURO J., RÖSSL C., THEISEL H.: Generalized metric energies for continuous shape deformation. In *Mathematical Methods for Curves and Surfaces*, vol. 8177 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2014, pp. 135–157. 16

[Mor01] MORITA S.: *Geometry of differential forms*. American Mathematical Society, Providence, R.I, 2001. 11

[MPP*13] MARCIAS G., PIETRONI N., PANOZZO D., PUPPO E., SORKINE-HORNUNG O.: Animation-aware quadrangulation. *Computer Graphics Forum 32*, 5 (2013). 3, 19

[MPZ14] MYLES A., PIETRONI N., ZORIN D.: Robust field-aligned global parametrization. *ACM Transactions on Graphics 33*, 4 (2014). 7, 18, 24

[MRMH12] MEHTA S. U., RAMAMOORTHI R., MEYER M., HERY C.: Analytic tangent irradiance environment maps for anisotropic surfaces. *Computer Graphics Forum 31*, 4 (2012). 17

[MWT11] MA C., WEI L.-Y., TONG X.: Discrete element textures. *ACM Transactions on Graphics 30*, 4 (July 2011). 17

[NPPZ12] NIESER M., PALACIOS J., POLTHIER K., ZHANG E.: Hexagonal global parameterization of arbitrary surfaces. *IEEE Transactions on Visualization and Computer Graphics 18*, 6 (2012). 16

[NRP11] NIESER M., REITEBUCH U., POLTHIER K.: Cubecover - parameterization of 3d volumes. *Computer Graphics Forum 30*, 5 (2011). 16, 24

[O'N66] O'NEILL B.: *Elementary Differential Geometry*. Academic Press, 1966. 4

[PBSH13] PANOZZO D., BLOCK P., SORKINE-HORNUNG O.: Designing unreinforced masonry models. *ACM Transactions on Graphics 32*, 4 (2013). 17

[PCCS11] PIETRONI N., CORSINI M., CIGNONI P., SCOPIGNO R.: An interactive local flattening operator to support digital investigations on artwork surfaces. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011). 17

[PCK*08] PARIS S., CHANG W., KOZHUSHNYAN O. I., JAROSZ W., MATUSIK W., ZWICKER M., DURAND F.: Hair photobooth: Geometric and photometric acquisition of real hairstyles. *ACM Transactions on Graphics 27*, 3 (Aug. 2008). 17, 19, 24

[Ped95] PEDERSEN H. K.: Decorating implicit surfaces. In *Proc. SIGGRAPH '95* (1995). 18, 19

[PFH00] PRAUN E., FINKELSTEIN A., HOPPE H.: Lapped textures. In *Proc. SIGGRAPH '00* (2000). 18, 19

[PHD*10] POTTMANN H., HUANG Q., DENG B., SCHIFTNER A., KILIAN M., GUIBAS L., WALLNER J.: Geodesic patterns. *ACM Transactions on Graphics 29*, 4 (2010). 17

[PL09] PENG Z., LARAMEE R. S.: Higher dimensional vector field visualization: A survey, 2009. 17

[PLPZ12] PANOZZO D., LIPMAN Y., PUPPO E., ZORIN D.: Fields on symmetric surfaces. *ACM Transactions on Graphics 31*, 4 (2012). 15, 19, 20

[PLS*15] PAN H., LIU Y., SHEFFER A., VINING N., LI C.-J., WANG W.: Flow aligned surfacing of curve networks. *ACM Transactions on Graphics 34*, 4 (2015). 17

[PP03] POLTHIER K., PREUSS E.: Identifying vector field singularities using a discrete Hodge decomposition. In *Visualization and Mathematics III* (2003). 7, 12, 15

[PPTSH14] PANOZZO D., PUPPO E., TARINI M., SORKINE-HORNUNG O.: Frame fields: Anisotropic and non-orthogonal cross fields. *ACM Transactions on Graphics 33*, 4 (2014). 10, 14, 16, 18, 19, 20, 21, 22, 24

[PS98] POLTHIER K., SCHMIES M.: *Mathematical Visualization*, vol. III. Springer Verlag, 1998, ch. Straightest Geodesics on Polyhedral Surfaces. 7

[PSP*14] PAL K., SCHÜLLER C., PANOZZO D., SORKINE-HORNUNG O., WEYRICH T.: Content-aware surface parameterization for interactive restoration of historical documents. *Computer Graphics Forum 33*, 2 (2014). 17

[PTP*15] PIETRONI N., TONELLI D., PUPPO E., FROLI M., SCOPIGNO R., CIGNONI P.: Statics aware grid shells. *Computer Graphics Forum* (2015). 3, 17

[PTSZ11] PIETRONI N., TARINI M., SORKINE O., ZORIN D.: Global parametrization of range image sets. *ACM Transactions on Graphics 30*, 6 (2011). 16

[PvdBC*11] PATIL S., VAN DEN BERG J., CURTIS S., LIN M., MANOCHA D.: Directing crowd simulations using navigation fields. *IEEE Transactions on Visualization and Computer Graphics 17*, 2 (2011). 17

[PZ07] PALACIOS J., ZHANG E.: Rotational symmetry field design on surfaces. *ACM Transactions on Graphics 26*, 3 (2007). 9, 19, 21, 24

[PZ11] PALACIOS J., ZHANG E.: Interactive visualization of rotational symmetry fields on surfaces. *IEEE Transactions on Visualization and Computer Graphics 17*, 7 (2011). 17, 18, 19, 24

[RGB*14] RAYMOND B., GUENNEBAUD G., BARLA P., PACANOWSKI R., GRANIER X.: Optimizing BRDF orientations for the manipulation of anisotropic highlights. *Computer Graphics Forum 33*, 2 (2014). 17

[RLL*06] RAY N., LI W. C., LÉVY B., SHEFFER A., ALLIEZ P.: Periodic global parameterization. *ACM Transactions on Graphics 25*, 4 (2006). 9, 15, 16, 19, 23

[RNLL10] RAY N., NIVOLIERS V., LEFEBVRE S., LÉVY B.: Invisible seams. In *Proc. Eurographics Symposium on Rendering* (2010). 16

[RS14] RAY N., SOKOLOV D.: Robust polylines tracing for n-symmetry direction field on triangulated surfaces. *ACM Transactions on Graphics 33*, 3 (2014). 7, 18

[RS15] RAY N., SOKOLOV D.: On smooth 3d frame field design. *CoRR* (2015). 11

[RVAL09] RAY N., VALLET B., ALONSO L., LÉVY B.: Geometry-aware direction field processing. *ACM Transactions on Graphics 29*, 1 (2009). 12, 14, 19, 23

[RVLL08] RAY N., VALLET B., LI W. C., LÉVY B.: N-symmetry direction field design. *ACM Transactions on Graphics 27*, 2 (2008). 5, 8, 9, 12, 14, 15, 19

[SBCBG11a] SOLOMON J., BEN-CHEN M., BUTSCHER A., GUIBAS L.: As-killing-as-possible vector fields for planar deformation. *Computer Graphics Forum 30*, 5 (2011), 1543–1552. 13, 16, 19, 24

[SBCBG11b] SOLOMON J., BEN-CHEN M., BUTSCHER A., GUIBAS L.: Discovery of intrinsic primitives on triangle meshes. *Computer Graphics Forum 30*, 2 (2011), 365–374. 13, 17

[Sch95] SCHWARZ G.: *Hodge Decomposition - A Method for Solving Boundary Value Problems*. Springer, 1995. 6

[SG71] STINY G., GIPS J.: Shape grammars and the generative specification of painting and sculpture. In *IFIP Congress (2)* (1971), pp. 1460–1465. 17

[SLCZ09] SPENCER B., LARAMEE R. S., CHEN G., ZHANG E.: Evenly spaced streamlines for surfaces: An image-based approach. *Computer Graphics Forum 28*, 6 (2009). 18

[TACSD06] TONG Y., ALLIEZ P., COHEN-STEINER D., DESBRUN M.: Designing quadrangulations with discrete harmonic forms. In *Eurographics Symposium on Geometry Processing* (2006), pp. 201–210. 10

[TGG06] TEWARI G., GOTSMAN C., GORTLER S. J.: Meshing genus-1 point clouds using discrete one-forms. *Computers & Graphics 30*, 6 (2006), 917–926. 10

[TPP*11] TARINI M., PUPPO E., PANOZZO D., PIETRONI N., CIGNONI P.: Simple quad domains for field aligned mesh parametrization. *ACM Transactions on Graphics 30*, 6 (2011). 16

[Tur01] TURK G.: Texture synthesis on surfaces. In *Proc. SIGGRAPH '01* (2001). 18, 19

[VBC15] VAXMAN A., BEN-CHEN M.: *Dupin Meshing: A Parameterization Approach to Planar Hex-Dominant Meshing*. Tech. Rep. CS-2015-01, Department of Computer Science,technion-IIT, 2015. 17

[vFTS06] VON FUNCK W., THEISEL H., SEIDEL H.-P.: Vector field based shape deformations. *ACM Transactions on Graphics 25*, 3 (2006). 11, 15, 16, 19

[VHWP12] VOUGA E., HÖBINGER M., WALLNER J., POTTMANN H.: Design of self-supporting surfaces. *ACM Transactions on Graphics 31*, 4 (2012). 17

[vTSSH13] VON TYCOWICZ C., SCHULZ C., SEIDEL H.-P., HILDEBRANDT K.: An efficient construction of reduced deformable objects. *ACM Transactions on Graphics 32*, 6 (2013), 213:1–213:10. 16

[vTSSH15] VON TYCOWICZ C., SCHULZ C., SEIDEL H.-P., HILDEBRANDT K.: Real-time nonlinear shape interpolation. *ACM Transactions on Graphics 34*, 3 (2015), 34:1–34:10. 16

[vW02] VAN WIJK J. J.: Image based flow visualization. *ACM Transactions on Graphics 21*, 3 (2002). 17

[War83] WARNER F. W.: *Foundations of differentiable manifolds and Lie groups*. Springer, 1983. 5, 6

[War06] WARDETZKY M.: *Discrete Differential Operators on Polyhedral Surfaces–Convergence and Approximation*. PhD thesis, Freie Universität Berlin, 2006. 7, 12, 13

[WBH*07] WARDETZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *Computer Aided Geometric Design 24*, 8-9 (2007). 7, 13

[Wij03] WIJK J. J. V.: Image based flow visualization for curved surfaces. In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)* (2003), VIS '03. 17

[WL01] WEI L.-Y., LEVOY M.: Texture synthesis over arbitrary manifold surfaces. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (2001), SIGGRAPH '01. 16, 19

[WTD*06] WANG K., TONG Y., DESBRUN M., SCHRÖDER P., ET AL.: Edge subdivision schemes and the construction of smooth vector fields. *ACM Transactions on Graphics 25*, 3 (2006). 10, 19

[WTHS04] WEINKAUF T., THEISEL H., HEGE H.-C., SEIDEL H.-P.: Topological construction and visualization of higher order 3D vector fields. *Computer Graphics Forum 23*, 3 (2004). 24

[WWH*14] WU X., WAND M., HILDEBRANDT K., KOHLI P., SEIDEL H.-P.: Real-time symmetry-preserving deformation. *Computer Graphics Forum 33*, 7 (2014), 229–238. 16

[YCLJ12] YAO C.-Y., CHI M.-T., LEE T.-Y., JU T.: Region-based line field design using harmonic functions. *IEEE Transactions on Visualization and Computer Graphics 18*, 6 (2012). 11, 17, 19

[YWVW13] YANG Y.-L., WANG J., VOUGA E., WONKA P.: Urban pattern: Layout design by hierarchical domain splitting. *ACM Transactions on Graphics 32*, 6 (Nov. 2013). 17

[Zha13] ZHANG E.: Npr for traditional artistic genres. In *Image and Video-Based Artistic Stylisation*, Rosin P., Collomosse J., (Eds.), vol. 42 of *Computational Imaging and Vision*. Springer London, 2013. 17

[ZHLB10] ZHANG M., HUANG J., LIU X., BAO H.: A wave-based anisotropic quadrangulation method. *ACM Transactions on Graphics 29*, 4 (2010). 15

[ZHT07] ZHANG E., HAYS J., TURK G.: Interactive tensor field design and visualization on surfaces. *IEEE Transactions on Visualization and Computer Graphics 13*, 1 (2007). 9, 10, 14, 17, 18, 19

[ZMT06] ZHANG E., MISCHAIKOW K., TURK G.: Vector field design on surfaces. *ACM Transactions on Graphics 25*, 4 (2006). 7, 18, 19

[ZP05] ZHENG X., PANG A.: 2d asymmetric tensor analysis. In *16th IEEE Visualization Conference (VIS 2005), 23-28 October 2005, Minneapolis, MN, USA* (2005), p. 1. 10

[ZSW10] ZADRAVEC M., SCHIFTNER A., WALLNER J.: Designing quad-dominant meshes with planar faces. *Computer Graphics Forum 29*, 5 (2010). 17, 19

[ZYLL09] ZHANG E., YEH H., LIN Z., LARAMEE R. S.: Asymmetric tensor analysis for flow visualization. *IEEE Transactions on Visualization and Computer Graphics 15*, 1 (2009). 10

[ZZCJ14] ZHUANG Y., ZOU M., CARR N., JU T.: Anisotropic geodesics for live-wire mesh segmentation. *Computer Graphics Forum 33*, 7 (2014). 17