Similarity Maps and Field-Guided T-Splines: a Perfect Couple

MARCEL CAMPEN and DENIS ZORIN, New York University

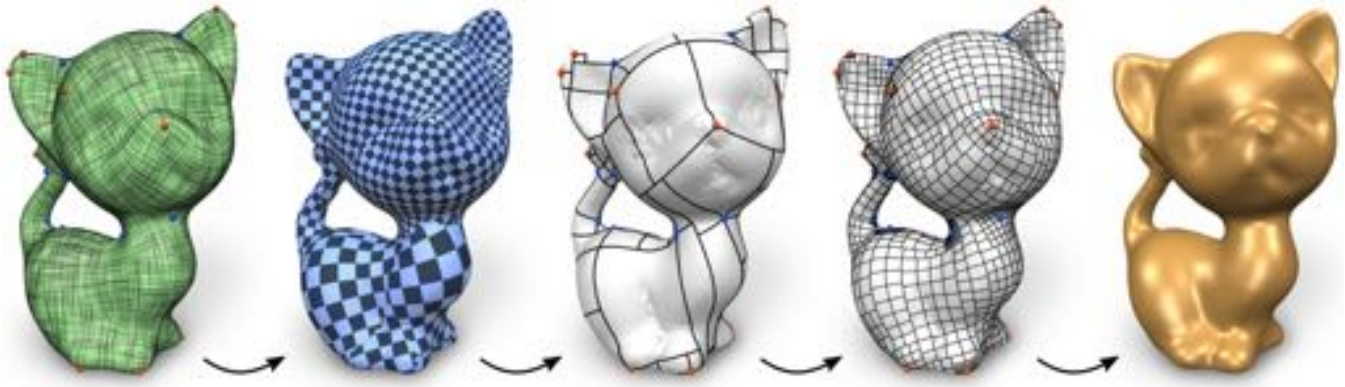


Fig. 1. Given a cross field (green) with arbitrary topological structure (in terms of extraordinary points and global holonomy), using a novel algorithm we compute a compatible global conformal parametrization with particular (*seamless similarity*) transitions (blue). Spline domains are derived from this global parametrization in the form of aligned T-meshes, and adapted to specific needs (gray). Using a novel halfedge-knots concept, a T-spline function space can be built on these base meshes and, for instance, be used to define, fit, and evaluate a smooth piecewise rational surface (golden). The construction is backward compatible with classical NURBS, T-Splines, and T-NURCCS, while enabling full algorithmic control over extraordinary points and global structure.

A variety of techniques were proposed to model smooth surfaces based on tensor product splines (e.g. subdivision surfaces, free-form splines, T-splines). Conversion of an input surface into such a representation is commonly achieved by constructing a global seamless parametrization, possibly aligned to a guiding cross-field (e.g. of principal curvature directions), and using this parametrization as domain to construct the spline-based surface.

One major fundamental difficulty in designing robust algorithms for this task is the fact that for common types, e.g. subdivision surfaces (requiring a conforming domain mesh) or T-spline surfaces (requiring a globally consistent knot interval assignment) reliably obtaining a suitable parametrization that has the same topological structure as the guiding field poses a major challenge. Even worse, not all fields do admit suitable parametrizations, and no concise conditions are known as to which fields do.

We present a class of surface constructions (T-splines with *halfedge knots*) and a class of parametrizations (*seamless similarity maps*) that are, in a sense, a perfect match for the task: for *any* given guiding field structure, a compatible parametrization of this kind exists and a smooth piecewise rational surface with exactly the same structure as the input field can be constructed from it. As a byproduct, this enables full control over extraordinary points. The construction is backward compatible with classical NURBS. We present efficient algorithms for building discrete conformal similarity maps and associated T-meshes and T-spline surfaces.

CCS Concepts: • **Computing methodologies → Mesh geometry models**

Additional Key Words and Phrases: T-mesh, T-NURCCS, seamless parametrization, conformal maps, holonomy, interval assignment

This work is supported by the National Science Foundation, under grant IIS-1320635.

ACM Reference format:
Marcel Campen and Denis Zorin. 2017. Similarity Maps and Field-Guided T-Splines: a Perfect Couple. *ACM Trans. Graph.* 36, 4, Article 91 (July 2017), 16 pages.
DOI: http://dx.doi.org/10.1145/3072959.3073647

1 INTRODUCTION

The topic of surface parametrization over planar domains has received considerable attention as it, very generally speaking, enables concepts from the Euclidean world to be employed on a Riemannian surface. Challenges arise when surfaces of arbitrary topology are to be dealt with, where such parametrizations globally inevitably exhibit discontinuities, or need to be defined chart-wise. Fortunately, if the concept to be employed on the surface is invariant to a certain class of parameter transformations, it can nevertheless be used obliviously if the parametrization is related across the discontinuities (the cuts or chart borders) by transitions from exactly this class.

An important transition class is that of translations composed with rotations by multiples of $\frac{\pi}{2}$. Perhaps somewhat misleadingly, the term *seamless parametrization* was coined for this case, *integer grid map* for the special case of integer valued translations. Such parametrizations, for instance, enable bivariate spline function spaces to be extended from the plane to manifold surfaces, or regular grids to be mapped to structured quad meshes on surfaces—due to certain translational and rotational invariances of spline basis functions and lattices, respectively (cf. Proposition 2).

The class of seamless transitions is small, constructing a global surface parametrization with transitions from only this class a challenge. While significant advances have been made in this regard in recent years, no reliable solution is known that ensures local injectivity and seamlessness while providing strict control over the parametrization's topological structure. This control is important

as it concerns such significant features as the parametrization's singularities (which, e.g., imply extraordinary points in spline surfaces or irregular vertices in quad meshes) as well as its global holonomy (cf. Section 2). The latter is of particular relevance when the global directional flow of the parametrization's isolines is sought to be influenced, e.g. using a guiding cross field (cf. Figure 1 left). Furthermore, even when a reliable, practical solution becomes available, it is known that not every cross field can be matched topologically by a seamless parametrization. This is discussed in detail in Section 2.

Contribution

We introduce two new concepts which, in tandem, provide a remedy.

- **Seamless similarity maps**. This class of parametrizations has transitions that are compositions of translations, rotations by multiples of $\frac{\pi}{2}$, *and* (positive, isotropic) scalings. It thus forms a superclass of seamless maps. This class is large enough that for *any* surface and *any* field topology a matching seamless similarity map exists. This statement even holds when we focus specifically on *conformal* seamless similarity maps (cf. Figure 3). This allows us to leverage results for conformal mapping and to present a practical algorithm for the construction of maps from this class, whose efficiency and robustness we demonstrate on a challenging benchmark.
- **T-splines with halfedge knots**. This construction is a remarkably simple generalization of classical T-splines: instead of assigning a single knot interval per edge of the control mesh, we assign two values, one for each side of the edge (subject to local conditions, cf. Figure 2). This removes a non-inherent restriction that limited applicability to domains built from the smaller class of seamless maps. These splines can thus be defined based on seamless similarity maps instead. We demonstrate that this, for instance, opens up possibilities for piecewise polynomial or rational smooth surface constructions with full structural control (over extraordinary points and global 'edge flow')

Herein we specifically focus on extending bi-cubic T-splines and their more generic version T-NURCCS [Sederberg et al. 2003], and demonstrate their use in the context of surface representation. Our surfaces are piecewise rational bi-cubic and C^2 away from extraordinary points, and C^1 at extraordinary points (or G^2 and G^1, respectively, in the terminology of geometric continuity). The approach is more general, though, and other degrees and similar function space constructions could be employed, as long as they support T-joints. Our generalization does not obstruct common local control mesh structure requirements such as well-separation of extraordinary vertices [Li et al. 2010; Sederberg et al. 2003] or non-intersecting T-extensions for analysis-suitable T-splines [Li et al. 2012].

Our algorithm for seamless similarity map construction extends conformal maps to surfaces of arbitrary topology, using a special type of cut conditions controlling global holonomy properties of the map. We believe use cases other than T-splines can benefit from this, possibly with some changes, as well. Our scenario does not require maps to be conformal; the initial map can thus be further optimized (within the class of seamless similarity maps), e.g. for other distortion measures and directional alignment, using established techniques with local injectivity constraints.

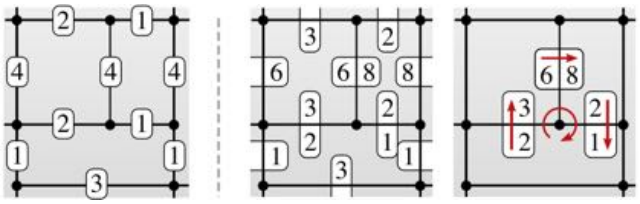


Fig. 2. Classically, knot intervals are assigned to edges of the spline control mesh (left). On opposite sides of each quad face, knot interval (sums) must be equal. We assign knot intervals per *halfedge* instead (middle). The *two* knot intervals thus assigned per edge may differ, which globally provides additional flexibility; only a *cycle condition* around each vertex needs to be fulfilled: the cyclic product of interval ratios across edges (right) must be 1, as in this example: $\frac{2}{3} \cdot \frac{6}{8} \cdot \frac{2}{1} = 1$.

2 BACKGROUND AND OVERVIEW

In order to define spline function spaces (or similar constructions) on surfaces, a global parametrization (that respects their respective invariances) is required. While in theory any such parametrization is sufficient, in practice it is often of significant benefit to use one of, in some sense, high quality. A method that generates a parametrization for this purpose should thus offer control over this.

A particularly important aspect in this regard are the (often unfavorable but topologically inevitable) extraordinary points, thus control over their number, types, and locations is desirable. Control over the directional orientation of parametric lines on the surface is likewise of relevance. For instance, alignment with respect to creases and principal curvature directions affects a spline surface's or mesh's quality [Alliez et al. 2003; Kovacs et al. 2015; Li et al. 2006]. Both, extraordinary points and desired directional orientations, can universally be specified in the form of a cross field (cf. Figure 1 left) serving as guidance for the parametrization [Vaxman et al. 2016].

2.1 Holonomy signature

The numbers of quarter rotations (by $\frac{\pi}{2}$) of a cross field (4-symmetry direction field) around arbitrary loops on the surface are invariant to local field perturbations; they describe the topology of the field [Ray et al. 2008]. Due to homotopy, it is sufficient to consider one contractible loop around each singularity and a pair of loops for each surface handle (cf. Figure 3) to completely understand the field's topology. We call the collection of these integer numbers the field's *(holonomy) signature*. Note that this signature is relative to the set of loops considered, and that on a genus 0 surface it only concerns the singularities. The signature of a seamless parametrization can be defined analogously, via the cross field formed by its parametric lines. Note that not every collection of numbers can occur: a cross

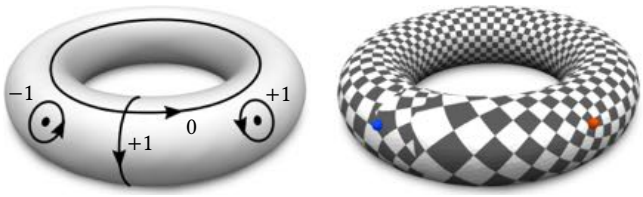


Fig. 3. Example of a holonomy signature for a genus 1 surface, with 2 singularities and 2 homology basis loops. Right: seamless similarity parametrization with this signature (singularities colored).

field (and thus its signature) fulfills the Poincaré-Hopf theorem [Ray et al. 2008]. We will tacitly make the standard assumption that the field has no non-hyperbolic singularities (with index ≥ 1).

It is essential that the signature of the constructed parametrization matches that of the given guiding field. If there is a different rotation number on a loop, there is no hope that (even with any form of local adjustment and optimization of the parametrization) good directional alignment to the field, or the desired local parametrization structure at extraordinary points, can be achieved. Constructing a parametrization that is topologically compatible (i.e. has the same signature) with a given cross field (or even just with its singularities) is an issue with available techniques:

(1) not all cross fields can be matched by a seamless parametrization; obstacles can be the singularity configuration ([Jucovič and Trenkler 1973] show that no such parametrization is compatible with the configuration in Figures 3 and 4) as well as the global holonomy structure (cf. Figure 5, as shown in [Myles et al. 2014, Appendix 1]).
(2) even if a compatible seamless parametrization exists, there is not necessarily a known method that can reliably find it. Also, there appears to be no complete understanding as to which holonomy signatures actually do admit such a parametrization.

By working with seamless *similarity* parametrizations instead, we are able to avoid these issues. This is illustrated in Figures 3, 4, and 5, where we visualize seamless similarity maps (using a checkerboard texture or derived T-meshes) computed for exemplary configurations that are problematic for seamless maps, as discussed above.

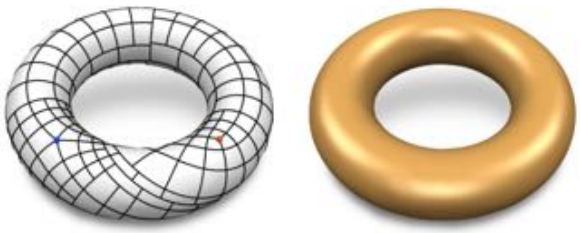


Fig. 4. While a conforming quad mesh (or seamless map) of genus 1 with a single valence 3 and a single valence 5 vertex is known not to exist, a T-mesh can be constructed (left). While this T-mesh does not allow for consistent edge knot intervals, using our halfedge knot intervals a T-spline surface can be defined and evaluated (right).

Fig. 5. T-spline surfaces (with superimposed images of the underlying control meshes) without extraordinary vertices. They differ in their holonomy signature (intuitively: the rotation of the control mesh 'edge flow' around the object). Only the left-most instance can be handled by seamless maps and classical T-splines with edge knots. Our variant with halfedge knots supports all three signatures (as well as any other).

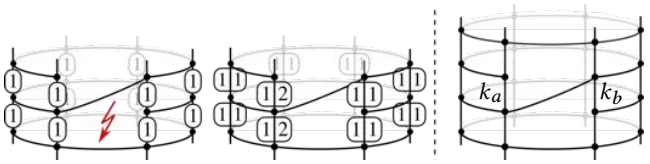


Fig. 6. Left: a cylindrical control T-mesh. It is not possible to assign positive knot intervals that are consistent in every face to the vertical edges. Middle: halfedge knot intervals can be assigned such that they are consistent in every face, as well as cycle consistent around every vertex. Right: on this mesh not even positive halfedge knot intervals can be assigned consistently; either per-face consistency needs to be violated, cycle consistency needs to be violated, or knot intervals k_a and k_b need to be set to zero.

2.2 Knot consistency

Bivariate tensor-product splines are generally defined with respect to an axis-aligned quad mesh in the parametric domain, in the case of T-splines a non-conforming one (a T-mesh) [Sederberg et al. 2003]. The associated knots determine the parametric extent of the individual quads. Vice versa, one can start by assigning extent values (knot intervals) to the edges of an abstract control mesh and let them implicitly define the underlying parametric domain and its knots. One, however, needs to ensure that the assigned knot intervals are *consistent*: on opposite sides of a quad the (sums of) intervals must be equal[1]; otherwise a corresponding axis-aligned domain rectangle does not exist [Kovacs et al. 2015; Sederberg et al. 2003].

Not every T-mesh allows for the assignment of globally consistent edge knot intervals—at least if we restrict ourselves to strictly positive values [Karciauskas et al. 2016]. Figure 6 left shows an example. Allowing zero intervals of course always leads to feasibility (just consider the trivial solution), but intervals may uncontrollably be forced to be zero, inadvertently lowering continuity.

Meshes (whether conforming or non-conforming) that are derived from (i.e. aligned with) a seamless map always allow for consistent positive knots: just notice that each aligned quad is mapped (modulo rigid transitions) to a rectangle by such maps. Hence, the parametric lengths of edges are, prototypically, knot intervals that are consistent.

Meshes aligned with a (more general) seamless similarity map, by contrast, do not always allow for consistent edge knot intervals. But one can more flexibly operate with knot intervals assigned per halfedge (cf. Figure 6) if local cycle conditions (cf. Figure 2) are fulfilled—which is always possible for meshes aligned with a seamless similarity map. The importance of the cycle conditions is discussed in Section 4.1.1.

While our generalization thus clearly enables a larger class of T-meshes to be used, we point out that *arbitrary* T-meshes (e.g. obtained by tracing in arbitrary cross fields) are not suitable for our construction: Figure 6 right shows an example where the per-face consistency conditions and the per-vertex cycle conditions cannot be met simultaneously with positive halfedge knot intervals.

[1] One can extend splines to support inconsistent knots [Müller et al. 2006; Sederberg et al. 2008], but this comes at the cost of losing compatibility with standard cubic NURBS; the surface is no longer piecewise rational or polynomial, or is of significantly increased degree—both of which may hamper interoperability in practical workflows.

Overview

We begin by showing the following results:

- Piecewise rational bi-cubic spline surfaces can be defined based on T-meshes with consistent halfedge knots (Section 4),
- T-meshes aligned with seamless similarity maps allow for such consistent halfedge knots (Section 5),
- Such a seamless similarity map exists for any cross field's holonomy signature (Section 6).

We then exploit these results algorithmically, following these steps in our construction, as illustrated in Figure 1:

(1) Compute a seamless similarity parametrization using a constrained conformal map, matching the desired holonomy signature. An optimization of this initial map for directional alignment and other objectives can follow (Section 7).
(2) Construct a suitable aligned control T-mesh, containing the set of desired spline control vertices, out of isoline segments of this parametrization and assign consistent halfedge knot intervals (Section 8).
(3) Evaluate the T-spline using basis function domains, and transition functions between these domains, that are derived from the consistent knot intervals (Section 4.2).

3 RELATED WORK

Seamless Parametrization. For global parametrizations to be of use in the context of generating parametric domains for classical tensor-product spline surfaces, they need to be seamless, as discussed in the previous sections. A variety of methods have been described for the generation of such parametrizations [Aigerman and Lipman 2015; Bommes et al. 2009; Campen et al. 2015; Ebke et al. 2016; Kälberer et al. 2007; Myles et al. 2014; Myles and Zorin 2012, 2013; Tong et al. 2006], often in the context of quad mesh generation or seamless texturing [Ray et al. 2010].

Conformal Parametrization. For the construction of seamless similarity parametrizations, we make use of conformal maps. There is a variety of approaches to conformal map construction, based on circle patterns [Bowers and Hurdal 2003; Kharevych et al. 2006], discrete Ricci flow [Chow and Luo 2003; Jin et al. 2007], or variations thereof [Springborn et al. 2008]. Conformal maps have been used in the concrete context of T-spline construction before [Gu et al. 2008; He et al. 2006]. These latter approaches are based on a specific mapping technique [Gu and Yau 2003] that supports extraordinary vertices of valence 8, 12, 16, . . . only.

Just like the above seamless metric parametrization techniques, these conformal techniques cannot typically guarantee finding a valid map in general. Most importantly, there can be obstructions to the existence of any proper map in the discrete setting, caused by unfit tessellation. Few works have addressed this problem, e.g. [Luo 2004], who shows that dynamic edge flips can theoretically be employed in the course of conformal flow.

We have not seen the global map behavior (in terms of holonomy or rotation around non-contractible loops) being addressed and controlled in these works (as opposed to some works on seamless parametrization, e.g. [Campen and Kobbelt 2014]). This control is an essential requirement and core aspect of our method.

It is well known that any surface admits *some* conformal atlas [Postnikov 2001, Ch. 14]. We show that this even holds for very specific conformal atlases, with seamless similarity transitions between charts. These build the foundation of our spline construction.

Smooth Surfaces. Smooth surface representations based on polynomial patches, NURBS, and, more generally, subdivision, were traditionally based on control meshes which are *conforming*, most prominently conforming quad meshes [Cashman 2012; Farin 2002], which are closely related to seamless parametrizations. Some works have addressed the problem of automatically constructing such meshes for given shapes, e.g. [Boier-Martin et al. 2004; Eck and Hoppe 1996; Panozzo et al. 2011]. Hierarchical control meshes later provided more freedom (in terms of local resolution adaptation capabilities) in this context [Bertram et al. 2004; Khodakovsky et al. 2000; Lounsbery et al. 1997; Zorin et al. 1997].

Even more flexibility was gained through constructions based on non-conforming quad meshes, so-called T-meshes. Examples are T-splines [Sederberg et al. 2004], T-NURCCS [Sederberg et al. 2003], PHT-splines [Deng et al. 2008], GPT-splines [Li et al. 2010], or dyadic T-subdivision [Kovacs et al. 2015]. Advantages due to this more flexible structure, e.g. for isogeometric analysis, have been demonstrated [Bazilevs et al. 2010].

Automatic T-Mesh Construction. The construction of arbitrary T-meshes can be facilitated quite easily, for instance by tracing integral curves of cross fields [Myles et al. 2014], by growing patches [Carr et al. 2006], or from isolines of periodic functions [Ray et al. 2006]. These T-meshes, however, are most often not suitable for spline applications, because they do not allow for globally consistent knot intervals (neither edge nor halfedge based).

T-meshes that allow for consistent edge-based knot intervals have been obtained as a submesh of a conforming mesh [Eppstein et al. 2008; Li et al. 2006; Myles et al. 2010; Sederberg et al. 2004], as a subset of the isocurves of a seamless surface parameterization [Campen et al. 2015; He et al. 2006], or by restricted local refinement as a supermesh of a conforming mesh [Sederberg et al. 2003; Wang et al. 2012]. However, robustly constructing a conforming quad mesh or a seamless surface map with the desired topology to start with is a major issue, as discussed in Section 2.1.

4 SPLINE SURFACES WITH HALFEDGE KNOTS

In this section, we define our version of T-spline surfaces with halfedge knot intervals in detail. To simplify exposition, we focus on surfaces without boundary and we do not consider zero intervals here (which can be of interest to selectively lower the local continuity order, e.g. for creasing).

T-meshes. A *T-mesh* is a mesh with quad faces; two quad faces of a T-mesh may intersect over *a part* of a common edge (as opposed to a conforming mesh, where faces only intersect over entire edges or vertices). A vertex is a *T-joint* with respect to an adjacent quad if it is contained in the interior of an edge of this quad. We define the *extended valence* of a vertex as the number of incident edges, plus the number of quads for which the vertex is a T-joint. We say that a vertex is *extraordinary* if its extended valence is different from four. All other vertices are called regular.

Halfedge knots. To define a T-spline we need to specify knot interval values. Here our construction differs from the classical one in a crucial way, as we do not assign a unique knot interval for an edge. Instead, we assign a knot interval per halfedge h, i.e., a (face, edge) pair. For a halfedge h of edge e, let h' denote the other (*opposite*) halfedge of e.

For each halfedge h of the T-mesh, corresponding to edge e and face f, we assign a knot interval value $k(h) = k(e, f) \in \mathbb{R}^{>0}$. These knot intervals need to satisfy two requirements:

- *the sum of the knot intervals of halfedges on a side of a quad is equal to the sum on the opposite side* (consistency condition),
- *the ratios $s_h = k(h)/k(h')$ of intervals of opposite halfedges satisfy $\prod_{h \in H_v} s_h = 1$ at every vertex v* (cycle condition).

Above, the set H_v contains one halfedge of each edge incident to v, chosen consistently (e.g. the ones associated with the faces that lie clockwise of the respective edges around v). Notice that this definition reduces to classical definitions of tensor-product spline and T-spline control meshes, which carry knot intervals per edge, if we require $s_h = 1$ for each halfedge h.

Admissibility. We call an assignment of halfedge knot intervals satisfying the above conditions *admissible*. A T-mesh is called admissible if an admissible assignment of positive knot intervals exists.

4.1 Spline definition

We associate each quad face Q_m of the T-mesh with a local coordinate system (u_m, v_m) and a parametric domain F_m (cf. Figure 7), of size given by the knot intervals of its halfedges. One corner of the quad is arbitrarily chosen as origin of F_m.

As with classical cubic B-splines or T-splines, each vertex v of the mesh is associated with a basis function (also called blending function) B_v and a control point $\mathbf{p}_v$. Each B_v is associated with a parametric domain D_v with a local coordinate system. We arbitrarily designate one of the quads incident to v its *anchor* quad Q_v, and choose the system of B_v to coincide with that of Q_v, except that its origin is located at the corner of v. As we will show, these choices of local systems are truly arbitrary; they do not affect the final result.

Just like for other basis splines, evaluation at an arbitrary point in a quad Q_m, i.e. at $(u, v) \in F_m$, is performed simply by evaluating each basis function at that point, and computing the accordingly weighted sum of their associated control points (cf. Section 4.2).

What needs our specific attention in this regard is the following:

- how do we transform between different coordinate systems, as B_v does not generally live in Q_m's local system?
- how is each B_v defined, i.e. how are their support regions and their knot vectors determined by the T-mesh?

The same questions arise for previous spline constructions with T-joints on manifolds; as the discussion in the literature is terse and our halfedge knots add another detail to be considered, we explain both aspects in detail in the following, and provide pseudo code in Supplement Appendix C to ensure complete clarity.

4.1.1 Transitions. For two edge-adjacent quads Q_1, Q_2 sharing an edge e, $\tau[F_1, F_2]$ is defined as the similarity transform mapping points on e in Q_1's coordinates to Q_2's coordinates, i.e. from F_1 to F_2 (cf. Figure 7). Note that the scaling factors s of τ are the halfedge knot ratios, and that the composition of these transitions around a regular vertex, due to the cycle condition, is the identity.

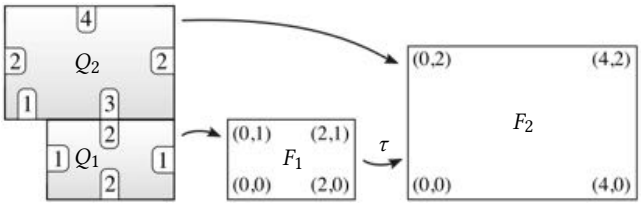


Fig. 7. Quads Q_m (gray) (with halfedge knots) and their associated parametric domains F_m (white). Here the transition is $\tau(F_1, F_2) = \frac{3}{2}R\mathbf{q} + (1, -\frac{3}{2})$, with a rotation R by angle zero in this case.

Let S_v denote the *supporting submesh* of B_v, consisting of those quads that are (fully or partially) overlapped by the support of B_v. To define the influence of a control point $\mathbf{p}_v$ (via basis function B_v) at a local parameter position (u_m, v_m) in a face $Q_m \in S_v$, i.e. to evaluate B_v at that position, we need to know the transform $\tau[F_m, D_v]$ that maps the F_m-coordinates (u_m, v_m) to D_v-coordinates. Then we are able to perform the required evaluation $B_v(\tau[F_m, D_v](u_m, v_m))$.

By definition, the coordinate systems of D_v and its anchor quad Q_v's domain F_v differ only by a (known) translation, defining $\tau[F_v, D_v]$. Through composition with the above transitions between quad domains, this can now be extended to other domains F. Concretely, we discover the entire supporting submesh S_v recursively, e.g. using a breadth-first traversal, starting from Q_v. For a quad Q_n adjacent to Q_v that is reached in this process, the transform $\tau[F_n, D_v]$ simply is $\tau[F_v, D_v] \circ \tau[F_n, F_v]$; for a quad Q_o adjacent to Q_n it is $\tau[F_n, D_v] \circ \tau[F_o, F_n]$, and so forth. The search stops at quads not contained in the supporting submesh S_v anymore.

This recursive definition of τ is independent of traversal order—if S_v is simply-connected and free of internal extraordinary vertices (which we will ensure for all relevant cases, cf. Section 4.1.3). Assume it was not well-defined, not path independent, i.e. there are two different directed paths π_1, π_2 from Q_v to some Q_o leading to different transforms $\tau_{\pi_1}[F_o, F_v] \neq \tau_{\pi_2}[F_o, F_v]$. These two paths, which are sequences of edges in the quad adjacency graph, together form a directed contractible cycle π_{12} enclosing only regular vertices, around which $\tau_{\pi_{12}} = \tau_{\pi_2} \circ \tau_{\pi_1}^{-1} \neq I$. Being contractible, this cycle π_{12} can be decomposed into directed elementary cycles (around single vertices), such that $\tau_{\pi_{12}}$ is the composition of the transitions around them. As $\tau_{\pi_{12}} \neq I$, at least around one of these regular vertices they must be non-trivial, contradicting our cycle condition assumption.

Remark. The above definition of τ between D and F extends to transitions between overlapping domains D_i, D_j in the obvious manner: $\tau_{ij} = \tau[D_i, D_j] = \tau[F_m, D_j] \circ \tau[F_m, D_i]^{-1}$, for any quad Q_m overlapped by D_i and D_j. Away from extraordinary points these transitions satisfy the usual chart cycle conditions $\tau_{jk} \circ \tau_{ij} = \tau_{ik}$, $\tau_{ij} = \tau_{ji}^{-1}$ for overlapping domains D_i, D_j and D_k. The manifold atlas with charts D_i thus defined, with similarity maps as transition maps, is a special case of a *similarity structure* on a surface.

4.1.2 Basis functions. It remains to determine the support and the knot vectors, and thus the size of the domain D_v, of a basis

function B_v. Note that these knots are relative to the local coordinate system chosen for B_v.

The process of computing the knot vectors is described in [Sederberg et al. 2003]: in each parametric direction (four at regular vertices, more or less at extraordinary ones), follow rays from v up to the (for the cubic case) second intersection with a vertex or orthogonal mesh edge. The parametric distances to the first and second intersections, together with the central zero, form the knots for B_v (cf. Figure 8 left). In our generalized case we merely need to make sure to apply the transition $\tau[F, D_v]$ for each quad domain F crossed by a ray (for rays coinciding with edges one can arbitrarily consider one of the two incident quads), so as to consistently measure distances in the common coordinate system of D_v. The extent of D_v, the parametric support of B_v, is defined by the range spanned by these knots.

Note that the supporting submesh S_v consists of those quads Q_m for which $\tau[F_m, D_v](F_m) \cap D_v \neq \emptyset$.

Invariance to choice of local systems. A different choice of anchor quad Q_v (thus local coordinate system) for basis function B_v leads to different knot vectors and a different domain D_v (and accordingly different transitions $\tau[F, D_v]$). The difference, however, is just a constant similarity transformation (e.g. for two adjacent possible anchor quads Q_m, Q_n this transformation is just $\tau[F_m, F_n]$ composed with a translation). For illustration, Figure 8 shows an exemplary T-mesh; for two different anchor quad choices (blue), the knot intervals mapped (using τ) into the respective basis function coordinate system are shown. Note that the two intervals associated with the two halfedges of an edge are equal in this common system (as the interval ratios are the scaling factors of τ). In this sense, admissible halfedge knots are *locally* equivalent to edge knots. Notice that between these two anchor choices the resulting intervals differ by a constant factor (2 in this case), as expected from a constant similarity transformation. As B_v is invariant to similarity transformation of knots and parameters (cf. Proposition 2), the resulting spline surface is indeed invariant to the choice of local system.

4.1.3 Extraordinary vertices. In the above we have generally assumed that $\tau[F_m, D_v]$ is well-defined for all F_m relevant to B_v. This holds if the supporting submesh S_v is simply-connected and free of internal extraordinary vertices. To help ensure these properties, and generally deal with extraordinary vertices in the following, we make the following additional assumptions on the connectivity of the T-mesh (and show in Section 8 that they can easily be met through local refinement operations):

A) Each extraordinary vertex is surrounded by three regular rings of conforming faces (no T-joints) with uniform knots.
B) For a regular vertex v the supporting submesh S_v does not overlap a face incident to an extraordinary vertex w, unless v is in the 1-ring or 2-ring of v.
C) Each S_v is simply-connected.

Herein we call an admissible T-mesh that meets these requirements *spline compatible*. We note that condition C is actually not essential but simplifies exposition and implementation (transitions τ would need to be defined in a per homotopy class manner to handle self-overlapping supports).

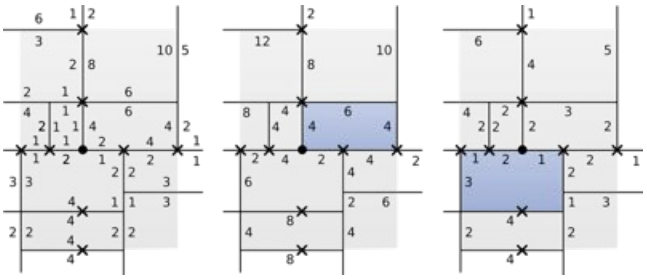


Fig. 8. Left: T-mesh with admissible halfedge knot intervals around a central vertex. Ray intersections that determine the knot vectors of the associated basis function are marked by crosses. Middle and right: the intervals mapped into a common coordinate system, for two different choices of anchor quads (blue). Only one interval per edge is depicted, because intervals of halfedge pairs coincide in the common system. Notice that the two cases (middle and right) differ only by a constant scale factor of 2.

Due to assumptions A and B, the only basis functions whose support contains an extraordinary vertex are the ones rooted at vertices incident to a face which is incident to an extraordinary vertex. The three possible configurations are depicted in Figure 9. To avoid having to handle numerous special cases, we exploit the symmetries of the basis function definition by conceptually splitting each basis function into quadrants (four for basis functions at regular vertices, more or less at extraordinary vertices), as indicated by colors in Figure 9. In the above constructions (Sections 4.1.1 and 4.1.2), each quadrant can be treated separately, as long as they are still associated with the same common control point $\mathbf{p}$.

The individual quadrant basis functions obviously have a smaller support; as can be observed in the figure, this leaves us with only a single case where the support (gray) of such a basis function contains an extraordinary vertex in its interior. In this case, the quadrant basis function is not well-defined. It can simply be ignored though, since it (in contrast to the other cases) overlaps *only* 1-ring faces of an extraordinary vertex—where the surface evaluation needs to be performed in a generalized manner (using subdivision rules) anyway. Condition A ensures that in these quads, where the usual tensor-product structure does not hold, the surface can be evaluated compatibly using uniform Catmull-Clark subdivision rules [Stam 1998] (or alternatively be capped [Peters 1992]). By adopting a more

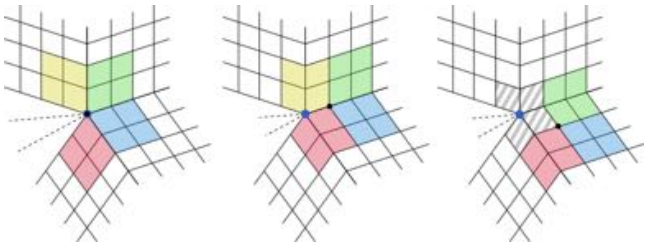


Fig. 9. Considering each quadrant of a basis function's support separately (different colors) allows to algorithmically handle regular vertices, extraordinary vertices, and regular vertices near extraordinary vertices in a unified way. In the case depicted on the right, one of the quadrants (in the gray 1-ring region) is irrelevant (and not well defined) because evaluation around extraordinary vertices is performed by special subdivision rules.

complex, non-uniform scheme, e.g. [Cashman 2012; Kovacs et al. 2015; Li et al. 2016; Sederberg et al. 2003], one could reduce the required number of regular rings to two.

4.2 Spline evaluation

To define precisely the evaluation of the surface at an arbitrary point in a quad Q_m (not incident to an extraordinary vertex), i.e. at $(u,v) \in F_m$, let C_m be the set of all overlapping quadrant basis functions B_i (for which $Q_m \in S_i$); let $\mathbf{p}_i$ be the control point associated with B_i. Then we define the surface evaluation as

$$\mathcal{F}(u,v) = \frac{\sum_{B_i \in C_m} \mathbf{p}_i \, B_i(\tau[F_m, D_i](u,v))}{\sum_{B_i \in C_m} B_i(\tau[F_m, D_i](u,v))} \tag{1}$$

For a quad incident to an extraordinary vertex, we rely on Catmull-Clark subdivision instead, to evaluate a C^1 surface [Stam 1998].

Proposition 1. *The surface $\mathcal{F}$ defined by* (1)

- *is invariant with respect to arbitrary seamless similarity transforms applied to domains D_i and their knot vectors, thus in particular to the choice of local coordinate systems,*
- *is a piecewise rational C^2 (or "G^2") surface away from extraordinary points.*

A proof can be found in Appendix A; it is based on the following observation of parametric similarity invariance of basis functions:

Proposition 2. *Let the 2D tensor-product B-spline basis function $B[\mathbf{u},\mathbf{v}](u,v) = B[\mathbf{u}](u)B[\mathbf{v}](v)$ be defined by knots $u_1,\dots,u_k$ and $v_1,\dots,v_k$; $\mathbf{u}$, $\mathbf{v}$ are vectors of knots for two coordinate directions. Let $(u',v') = \tau(u,v) = sR[u,v]^T + \mathbf{t}$ be a similarity transformation of the parametric domain B, where R is a rotation by $k\pi/2$, $k \in \mathbb{Z}$, $s > 0$ is a scale factor, and $\mathbf{t}$ a translation vector. Define $\mathbf{u}'$ and $\mathbf{v}'$ as vectors of knots u'_i and v'_i obtained as coordinates of $\tau(u_i, v_i)$. Then*

$$B[\mathbf{u}',\mathbf{v}'](u',v') = B[\mathbf{u},\mathbf{v}](u,v) \tag{2}$$

A proof can be found in Appendix A as well.

5 SPLINE-COMPATIBLE T-MESHES FROM SIMILARITY MAPS

To simplify the seemingly complex problem of constructing an admissible (or ultimately spline-compatible) T-mesh for a surface $\mathcal{M}$ (the conditions on knot intervals are nonlinear), we first reduce it to the problem of finding a seamless similarity map. The construction of such a map is then addressed in the following Sections 6 (theoretical background) and 7 (computational algorithm).

Consider a surface M with an embedded set of curves that cut it to a disk (*cut graph*). Denote the resulting cut surface with boundary M_c. A continuous, locally injective map $f(\mathbf{p}) : M_c \to \mathbb{R}^2$ is a seamless similarity map if for each two coinciding points $\mathbf{q}$, $\mathbf{q}'$ on different sides of the cut the parametric coordinates are related by a seamless similarity transformation, i.e. $f(\mathbf{q}') = \sigma(f(\mathbf{q})) = sRf(\mathbf{q}) + \mathbf{t}$, where R is a rotation by $k\pi/2$, $k \in \mathbb{Z}$, and these transitions σ fulfill the above mentioned cycle condition around any regular point. Note that this in particular implies that s, R, and $\mathbf{t}$ are constant along each branch of the cut graph (away from extraordinary points).

The proof of the following proposition is constructive, and defines a way to generate T-meshes and assign admissible halfedge knot intervals from a seamless similarity parametrization.

Proposition 3. *For a seamless similarity parametrization f on M_c, any T-mesh embedded in M whose edges follow f's parametric lines and whose vertices include f's singularities is admissible, and such a T-mesh always exists.*

Proof. An algorithm demonstrating existence is given in [Myles et al. 2014]: by tracing a general cross-field on a surface (in our special case: the field defined by the parametric lines of f) starting from extraordinary points, one can obtain an embedded T-mesh with the required properties, using a modified motorcycle graph algorithm [Eppstein et al. 2008]. As in our case parametric lines rather than general cross-fields need to be traced [Campen et al. 2015], in practice tracing is particularly simple compared to the general setting of [Myles et al. 2014; Ray and Sokolov 2014].

For the first part of the proposition, assume the cut coincides with edges of an embedded T-mesh, i.e. the cut nowhere runs through faces. We assign as knot interval value to each halfedge its length in parameter space, under f. As each face has a rectangular image (due to f-aligned edges), these knot intervals are consistent per face. Furthermore, the knot ratio at each edge equals the scale factor of the similarity transition σ on the cut coinciding with the edge (or is 1 if there is no cut). Hence, as f is cycle consistent, so are the knot intervals, proving admissibility of the T-mesh.

If the cut does not coincide with T-mesh edges, it can be deformed to do so; then the above argument applies. This is possible in general because the embedded T-mesh forms a cell decomposition of M (due to all faces having disk topology), thus is homotopic to M. □

In practice, such a cut deformation does not need to be performed explicitly. When computing parametric lengths of halfedges in a face, one simply maps all parameter values into a common local system (e.g. rooted at a corner of the face) by applying all cut transitions σ encountered on a path (within the face) to that corner. Cycle consistency of the map together with simple topology of the face ensure that the result is unique, path-independent.

6 SIMILARITY MAPS VIA CONFORMAL MAPS

In this section we establish existence of seamless similarity maps with arbitrary signature, using conformal maps. To this end, let us first make precise the notion of a cross field and its signature. For a background on the (discrete) exterior calculus notation we make use of, we refer to [Abraham et al. 1988; Desbrun et al. 2008].

Cross field. An assignment of quadruples $(\mathbf{e}, \mathbf{e}_\perp, -\mathbf{e}, -\mathbf{e}_\perp)$ of unit-length vectors, separated by right angles, to all points on a surface, possibly excluding a number of isolated extraordinary points (singularities), is called a cross field. It has an associated differential quantity, the *connection 1-form* ω (from which it can be recovered up to a constant global rotation). For the discrete case this viewpoint was considered in [Crane et al. 2010]. The form ω is defined by $\langle \omega, \mathbf{v} \rangle = \mathbf{e}_\perp \cdot \nabla_{\mathbf{v}} \mathbf{e}$, where $\nabla_{\mathbf{v}}$ is the surface derivative of $\mathbf{e}$ in the direction $\mathbf{v}$. This form ω, evaluated on some tangent vector $\mathbf{v}$, produces the *rate of rotation* of the cross field in direction $\mathbf{v}$. The total rotation r_γ along a curve γ (with tangent $\mathbf{t}$) is thus given by

$$r_\gamma = \int_\gamma \langle \omega, \mathbf{t} \rangle \, ds \tag{3}$$

The connection 1-form ω of a cross field is trivial [Crane et al. 2010], in the sense that this total rotation r_γ plus the total geodesic curvature $\kappa^{tot}[\gamma]$ of the curve itself is a multiple of $\pi/2$ for any *closed* curve (i.e. loop) γ. Note that the values $k_\gamma = (r_\gamma + \kappa^{tot}[\gamma])/\frac{\pi}{2}$ are the integral quarter rotation numbers that make up the signature introduced in Section 2.1. For a seamless (similarity) map f, the signature can be defined analogously, via the cross-field (and its differential ω_f) defined by the tangents to the parametric lines.

Proposition 4. *For any admissible holonomy signature for a system of homology loops γ_j, $j = 1 \dots 2g$ and singularities $\mathbf{p}_i$, $i = 1 \dots N$ on a surface M, there is a seamless similarity map with this signature; one such map is given by a conformal map f on the cut surface M_c, defined by the conformal factor e^ϕ on M_c, constructed as $d\phi = \star\omega_f$ from the unique co-closed 1-form ω_f on M satisfying*

$$\star d\omega_f = K - K_G, \tag{4}$$

$$\int_{\gamma_j} \langle \omega_f, \mathbf{t} \rangle \, ds = k_{\gamma_j} \frac{\pi}{2} - \kappa^{tot}[\gamma_j], \tag{5}$$

for each $j \in \{1, \dots, 2g\}$. K_G is the Gaussian curvature of M, and $K = 2\pi \sum_{i=1}^{N} index(\mathbf{p}_i)\, \delta(\mathbf{p}_i)$, with δ a Dirac delta distribution.

Equation (4) is effectively the limit case of (5) for loops contracting around every single point of M, thus, besides enforcing the desired rotation numbers for singular points $\mathbf{p}_i$, also enforcing an index of zero for any non-singular point [Bunin 2008].

We point out that this proposition in this form applies to the smooth case only; the discrete setting is considered in Section 7.

Proof. Let ϕ' be a 0-form on the closed surface M that solves the Poisson equation $\Delta\phi' = \star d \star d\phi' = K - K_G$. This equation has, for the here relevant case of K being a sum of deltas, a unique (up to an additive constant) solution ϕ' [Aubin 1998; Bunin 2008; Troyanov 1991]. Note that $\omega' = \star d\phi'$ is co-closed and obviously solves (4)—but not generally (5).

A closed genus g surface has a $2g$-dimensional space of harmonic 1-forms. Let $\{\omega_1, \dots, \omega_{2g}\}$ be a basis of this space, such that $\int_{\gamma_j} \langle \omega_i, \mathbf{t} \rangle \, ds = \delta_{ij}$, i.e. the total rotation of ω_i vanishes on all loops but γ_i [Gu and Yau 2003; Tong et al. 2006, App. B]. Then we have $d(\omega' + c_i\omega_i) = d\omega'$ for any $c_i \in \mathbb{R}$ due to harmonicity of ω_i, and $\int_{\gamma_j} \langle (\omega' + c_i\omega_i), \mathbf{t} \rangle \, ds = \int_{\gamma_j} \langle \omega', \mathbf{t} \rangle \, ds + c_i \int_{\gamma_j} \langle \omega_i, \mathbf{t} \rangle \, ds$. Hence, by choice of coefficients c_i, we can independently adjust the rotation around the loops γ_j, without affecting (4), such that $\omega_f = \omega' + \sum_{i=1}^{2g} c_i\omega_i$ uniquely solves (4) and (5).

As $\star\omega_f$ is closed, it is exact on the (simply-connected) cut surface M_c, so it can indeed by integrated uniquely (up to constant global scale), yielding the 0-form ϕ on M_c. Assuming the singular points are contained in the cut, M_c is flat under the metric $Ge^{2\phi}$ whose Gaussian curvature is K [Aubin 1998; Schoen and Yau 1994], defining a conformal map f. It is unique up to a rigid transformation. As discussed in [Myles and Zorin 2013], ω_f is indeed the differential of f constructed in this way. Thus f respects the input signature.

It remains to show that f is a seamless similarity map. For this we refer to Appendix A.

□

7 DISCRETE SIMILARITY MAPS

We now describe our algorithm to construct a discrete seamless similarity map on a closed orientable triangle mesh M, respecting a given signature. It is based on Proposition 4, although we do not follow the constructive existence proof, but solve the equation system (4) + (5) directly. This allows us to deal with discretization issues (e.g. mesh connectivity changes) in a simpler, unified manner.

7.1 Background

Solving this system (to then obtain ϕ via $d\phi = \star\omega_f$) is similar to other conformal mapping approaches, e.g. [Ben-Chen et al. 2008; Springborn et al. 2008], though the focus typically is on simply connected domains and thus (4) only. We need to employ an approach that supports taking (5) into account as well. We have chosen to design an algorithm building on multiple ideas from previous work, combining them in a novel way for the sake of generality, performance, and robustness.

We combine three main ideas: (1) the idea of defining a discrete flow evolving the discrete 0-form ϕ (or, in our case, actually its differential 1-form $\xi = -\star\omega_f$) according to deviation from the desired curvature [Luo 2004], (2) performing triangle mesh connectivity modifications as the flow proceeds, to enable convergence to a valid solution in the discrete setting [Gu et al. 2013; Luo 2004], and (3) discretization of a single flow step following a linear formulation [Ben-Chen et al. 2008] which corresponds to choosing the descent direction indicated by Newton's method for the flow [Springborn et al. 2008]. The resulting combination passes challenging stress tests as detailed in Section 9, and, in cases that do not require numerous mesh modifications, proves to be efficient.

Setting. Given is M with discrete metric G, given by edge lengths, and an admissible holonomy signature, e.g. one derived from a cross field. The output is a piecewise-linear seamless similarity map f on M' (a possibly modified or refined version of M). This map f is discrete conformal, but as we discuss in Section 7.4, it can be taken as a starting point (with the correct signature, i.e. topology) for optimization for other distortion measures, alignment, etc.

7.2 Solving for the 1-form ξ

We set up a system of linear equations discretizing (4) and (5). As variables we use the discrete 1-form ξ, i.e. a scalar value ξ_i per oriented edge i of the triangle mesh (the reversely oriented edge, i', implicitly has the negated value, $\xi_{i'} = -\xi_i$), and define $\omega = \star\xi$. Let the loops γ_j, $j = 1, \dots, 2g$, in the discrete case cyclic triangle strips, be represented by the edges between successive triangles in the strip, oriented consistently from left to right side of the strip. Following [Crane et al. 2010; Myles and Zorin 2013] the integral in (5) is discretized as $\int_\gamma \langle \star\xi, \mathbf{t} \rangle ds = \sum_{i \in \gamma} \star\xi_i$, and the Hodge star $\star$ is defined as $\star\xi_i = \frac{1}{2}(\cot\alpha_i + \cot\alpha_{i'})\xi_i$, where α_i is the angle opposing oriented edge i in the triangle left of i. The total geodesic curvature $\kappa^{tot}[\gamma]$ is easily computed by summing up turning angles along the path [Myles and Zorin 2013]. Thus, for each cycle γ_j, $j = 1, \dots, 2g$, and each vertex (except one), we setup equations

$$\sum_{i \in O_v} \star\xi_i = 2\pi - k_v \frac{\pi}{2} - K_G[v], \quad \sum_{i \in \gamma_j} \star\xi_i = k_{\gamma_j} \frac{\pi}{2} - \kappa^{tot}[\gamma_j], \tag{6}$$

where O_v is the set of outwards oriented edges incident to vertex v, $K_G[v]$ is the discrete Gaussian curvature at v, and k_v is the quarter rotation number around this vertex (which is related to the singularity index as $\text{index}(v) = 1 - k_v/4$).

As ξ is required to be a *closed* 1-form, we furthermore add closedness conditions, i.e. for each face (except one) with (counterclockwise oriented) edges i, j, k:

$$\xi_i + \xi_j + \xi_k = 0. \tag{7}$$

The one vertex and face that are skipped are chosen arbitrarily (closedness and correct curvature (via Gauss-Bonnet theorem) are implied there). The resulting linear equation system $A\xi = b$ has $2g + |V| - 1 + |F| - 1$ equations, which, according to Euler's formula, equals the number of variables, $|E|$. The equations are linearly independent, as a consequence of independence of loops (cf. [Ray et al. 2008]), so the system has a unique solution ξ.

Due to closedness, ξ is integrable (*exact*) on a simply-connected cut mesh M_c. Integration of ξ yields a logarithmic scale factor ϕ on M_c, defined uniquely up to an additive constant. ϕ defines a discrete metric G' [Luo 2004; Springborn et al. 2008] under which an edge (v, w) that has length ℓ under G has length

$$\ell' = e^{(\phi_v + \phi_w)/2} \ell \tag{8}$$

There are two problems with this linear method, cf. [Ben-Chen et al. 2008; Springborn et al. 2008]: the resulting discrete metric G' is not exactly flat (away from singularities), due to discretization, and the resulting edge lengths ℓ' may violate triangle inequalities. We resolve these problems by the algorithm described in the following, which employs the above linear solve iteratively, updating the system at each step using the new metric, and applying adaptive triangulation modifications, to maintain triangle inequalities. We found the remeshing capability of our method to be critical (cf. Table 1, column *edgeflips*), which is supported by the data of [Chien et al. 2016] on the low success rate of a non-remeshing discrete conformal parametrization approach on a benchmark data set.

7.3 Iterative Algorithm

The algorithm evolves the discrete metric G', i.e. the edge lengths of the cut mesh M_c. By $\phi(\xi)$ we denote ϕ obtained by integrating ξ on M_c, and method rescale updates edge lengths as in (8).

$G' \leftarrow G$

while $||b(G')||_\infty > \varepsilon$

  $\xi \leftarrow A(G')^{-1} b(G')$

  $\lambda \leftarrow \text{firstDegeneracy}(M_c, G', \xi)$

  $G' \leftarrow \text{rescale}(G', \lambda\phi(\xi))$

  For each triangle f degenerate under G': flip(f)

Effectively, we view the evolution as piecewise-linear flow: at every step, we attempt to linearly evolve the metric from the current metric G' to the rescaled metric with $\phi(\xi)$ obtained from the linear solve $A\xi = b$, stopping to modify the triangulation at intermediate points where a degeneracy event occurs.

7.3.1 Degeneracy handling. Function firstDegeneracy(M, G', ξ) determines the first triangle degeneration event (if any) when continuously evolving the metric as rescale$(G', \lambda\phi(\xi))$, i.e. it determines the smallest "time" $\lambda \in (0, 1]$ for which there is one (or more) degenerate triangle in M. Degeneration events are determined as the roots of an indicator function $I_{ij,k}(\lambda)$ per edge (i, j) of each triangle (i, j, k):

$$I_{ij,k}(\lambda) = \ell_{jk} e^{\xi_{ik}\lambda/2} + \ell_{ki} e^{\xi_{jk}\lambda/2} - \ell_{ij}, \tag{9}$$

where edge lengths ℓ are with respect to current metric G'. It has these properties: (1) for a triangle valid at λ, all three indicator functions are positive; (2) for a degenerate triangle, the indicator function of (at least) one edge vanishes; (3) for a triangle violating the triangle inequality, one indicator function is negative; (4) the function has at most two roots in λ, as shown in Appendix B.

We initially set $\lambda_0 = 1$. Then, for an indicator function I with $I(\lambda_0) < 0$ (indicating a violation at time λ_0), we find the root r in the range $(0, \lambda_0)$ and set $\lambda_0 = r$. Note that due to $I(0) > 0$ and property (4) there is exactly one root in this range. This is repeated until all $I(\lambda_0) \geq 0$, so as to determine the earliest degeneracy time λ_0, which is then returned by firstDegeneracy.

The method flip(f) is used to cure the degenerate triangle f. Let e be the longest edge of triangle f. Note that this edge is unique, as there are no edges of length zero at any step. Edge e is flipped and its length computed *intrinsically* from the incident triangles, as described in detail by [Fisher et al. 2007]. Notice that triangles can never degenerate to a point (a situation that could not be resolved by an edge flip) as the factors $e^{(\phi_v + \phi_w)/2}$ (cf. Equation (8)) applied to edge lengths in each iteration are strictly positive for any choice of ϕ. Figure 10 demonstrates the effect of this strategy.

Remark. One can perform flips more generally to make the triangulation intrinsically Delaunay [Fisher et al. 2007] after each iteration. This can speed up the process in cases where otherwise many degeneracy events would occur (cf. Table 1). Furthermore, if required, in the end all edge flips performed can be realized as edge splits instead (by overlaying the meshes [Fisher et al. 2007]), ensuring the output mesh is a refinement of the input mesh.

7.3.2 Analysis and relation to other methods. Discrete Ricci or Yamabe flow [Luo 2004] can be viewed as a continuous gradient descent of some convex energy $E(\phi)$ (in some instances called Ricci energy or Yamabe energy), which describes the deviation from the prescribed curvature (the right-hand side of (4)) under the metric implied by ϕ. In practice very small step lengths need to be chosen

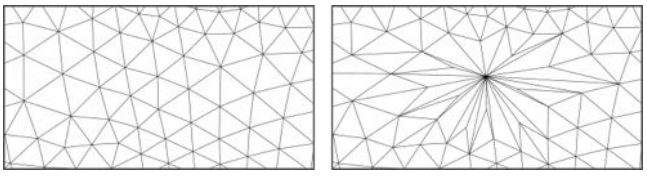


Fig. 10. Extreme example of automatic triangulation modification to support a high valence cone. A cone of 14π, corresponding to an extraordinary T-mesh vertex of valence 28, was prescribed at the central vertex of the input triangle mesh, which had valence 6, which implied it cannot support this cone (a total angle of 14π in the parametric domain). Remeshing has increased it valence so that 14π becomes representable.

for a proper first order discretization of this process. Significantly faster convergence can be achieved by using a second order, Newton method [Springborn et al. 2008]. Convergence from any starting point (any initial curvature/metric G) is guaranteed when using a globally convergent variant (e.g. with adaptive step size based on trust-region or line-search techniques) [Springborn et al. 2008]. Our algorithm can be interpreted as a variant of this approach. While in that approach the Newton update ϕ in each step is essentially determined via the linear system $(\star d \star d)\phi = \mu b$ (for an appropriately determined step size μ), our core iteration effectively just uses a different bracketing $(\star d \star)(d\phi) = \mu b$ (and exploits the additional degrees of freedom to add terms (5) for non-contractible loops to this underdetermined system) to determine the update for ξ $(= d\phi)$.

An extension of the convergence analysis of [Springborn et al. 2008] to this more general setting and a proof that in the case of convergence the resulting discrete conformal metric does indeed yield a discrete seamless similarity map with the desired signature can be found in [Campen and Zorin 2017]. The main open question is whether an infinite sequence of edge flips could occur and hinder convergence [Luo 2004]. The observed behavior, demonstrated in Section 9, has not yet indicated this.

7.4 Alignment Optimization

As T-mesh edges are aligned with isolines of the conformal map (to guarantee admissibility), the map's orientation (in terms of isoline directions) directly determines T-mesh edge orientation. So far, our focus was entirely on structure control: the map we are constructing has exactly the signature of a guiding cross-field, but it may be misaligned with the actual directions of the field. For some applications this may not be of concern, but for others, like spline fitting, such field alignment can certainly be of interest (cf. Section 2).

A very simple first adjustment one can make is applying a globally constant rigid rotation to the map, such that its isolines align, as well as rigidly possible, with the given guiding cross field.

To compute a further improved version of the map, we propose to set up an optimization problem based on the quadratic cross field alignment objective of [Bommes et al. 2009], with conservative convex local injectivity constraints (as described by [Lipman 2012] or [Bommes et al. 2013]), and linear constraints that fix the transitions across cuts, thereby preserving seamless similarity.

The injectivity constraints are initialized (convexified) with respect to the initial conformal map's parametric directions, insuring feasibility. The constraints fixing the transitions are of the form $(v - w) = sR(v' - w')$ (for each triangle mesh edge on the cut), where v and w are (the parametric coordinates of) the two vertices of a cut edge, and v', w' their duplicates on the other side of the cut. The scale factor s and the rotation R are read off the initial conformal map. Again, the initial map clearly satisfies these constraints, guaranteeing feasibility.

The important key aspect here is that the initial conformal map we computed provides a *feasible* solution (thus a valid starting point for optimization) to the problem of finding a locally injective seamless similarity parametrization. Finding such a feasible solution is usually the main challenge for the above methods in this context, which is solved by our conformal map initialization.

An example result of this optimization is shown in Figure 11.

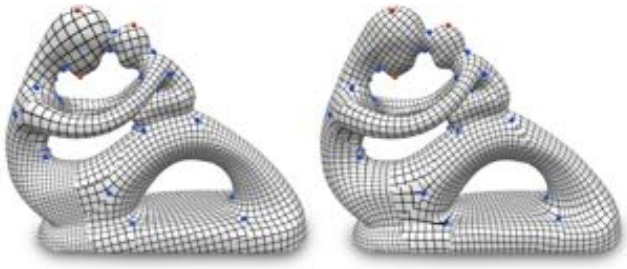


Fig. 11. Left: seamless similarity parametrization, rotated for (as good as rigidly possible) principal direction alignment. Right: optimized, deformed map with improved alignment. Notice that the similarity transitions across the cuts are preserved.

8 CONSTRUCTING THE T-MESH

Given a seamless similarity parametrization of the surface M, it is easy to construct an admissible T-mesh on M. As Proposition 3 shows, it is sufficient to construct *any* T-mesh with edges following the parametric lines of the similarity parametrization and vertices at singularities. Its proof gives a constructive way of defining halfedge knot intervals on such a T-mesh, and shows that this knot assignment is admissible.

A particularly coarse partition that fulfills the conditions can be obtained as the parametrization's motorcycle graph on M. It is constructed by simultaneously tracing isolines from the singularities (applying transition functions on the way so as to be independent of cuts) [Campen et al. 2015]. The graph forms a homotopic cell decomposition of M, whose cells are four-sided and aligned with isolines by construction [Eppstein et al. 2008]. Figure 14 left and Figure 20 show examples.

Once such a coarse T-mesh is available as starting point, one can easily perform mesh modifications, such as face splits or parallel edge shifts [Myles et al. 2010] to adapt it to specific needs, e.g. refine it to a desired resolution (possibly adaptively, fitting error driven [Li et al. 2006]). Unfortunately, the initial motorcyle graph, while having a low number of faces, can contain some small, and in particular long but narrow faces, making it not well suited for such refinement strategies. We thus apply a sequence of simple edge removals and "edge flips" to the initial mesh, with the goal of greedily maximizing the minimal face aspect ratio. This is detailed in the following.

8.1 T-Mesh Regularization

It is important to note that we cannot perform arbitrary mesh modifications; we need to preserve admissibility of the mesh. This is guaranteed if we just keep edges aligned to isocurves, preserve four-sidedness, and do not touch extraordinary vertices. We make use of the following operations that respect this.

Narrow faces can be removed by merging them with a neighboring face. This entails the removal of the separating edge(s) (cf. Figure 12 left). An edge can, however, only be removed if four-sidedness is preserved. Edges violating this can be removed anyway, if we insert another (orthogonal) edge instead, establishing four-sidedness again (cf. Figure 12 right). This can be interpreted as an edge flip.

Furthermore, we do not want to remove edges incident to extraordinary vertices as this would complicate processing in later stages.

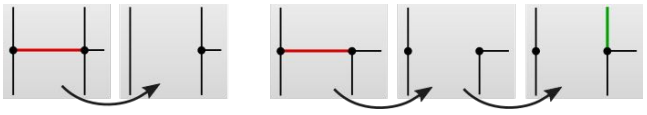


Fig. 12. Left: an edge (red) that can easily be removed. Right: an edge (red) whose removal leaves a non-foursided face (with a concave, L-joint corner). This is remedied by inserting another edge (green) at the corner.

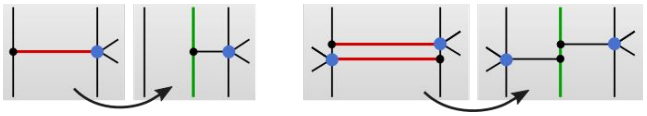


Fig. 13. Partial edge (red) removal with orthogonal edge (green) insertion at extraordinary vertices (blue) where edges cannot be removed entirely.

Such edges can, however, be removed partially, as illustrated in Figure 13 left. Again, orthogonal edges (green) are inserted to ensure four-sidedness. While partial removal might seem to be of limited use, it in particular enables the removal of narrow faces commonly found around extraordinary vertices, as depicted in Figure 13 right.

We apply these operations greedily in order of decreasing aspect ratio of the incident faces, i.e. starting at the worst-shaped face. An operation is only executed if it does not introduce new faces of worse aspect ratio. Example results of this process are shown in Figure 14 and in Figure 20.

8.2 Spline Compatibility

To make a T-mesh constructed in any of the above ways suitable for the definition of a spline surface, we need an algorithm to modify an admissible T-mesh into a spline compatible T-mesh, fulfilling property A, B, and C.

Condition C. is easily met by locally refining the mesh, splitting faces in supports which are initially of non-disk topology.

Condition A. (three regular rings) is fulfilled by determining the parametric "clearance" c (the parametric length of the shortest incident edge) around an extraordinary vertex v and inserting three regular rings of quads of parametric size $\frac{1}{4}c$ around v (cf. Figure 15 middle). As this initially leaves non-foursided faces, one edge is extended at each outer corner of the 3-ring, splitting them into

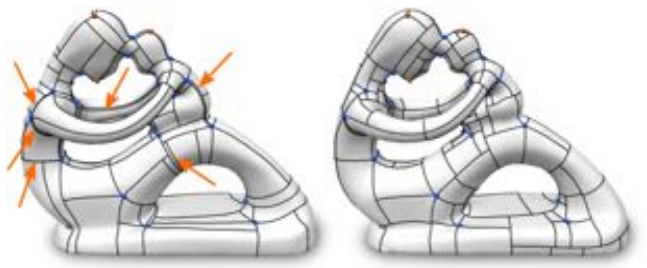


Fig. 14. Left: initial motorcycle graph T-mesh; numerous narrow patches with aspect ratios up to 1:471. Right: using greedy edge removal and edge flip operations, the faces can be made significantly squarer (aspect ratios at most 1:5.5).

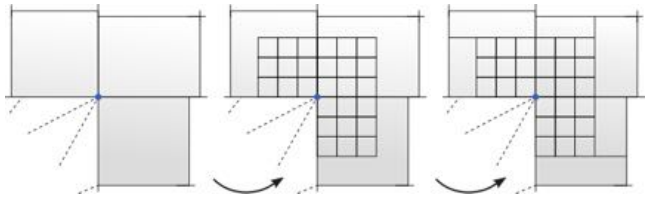
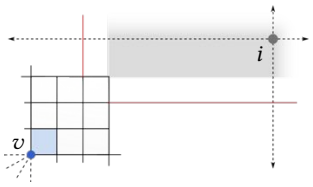


Fig. 15. Refinement around an extraordinary vertex (blue) of arbitrary valence (left) to ensure three regular and uniform rings of quads (middle) and preserve four-sidedness of all faces (right).

two four-sided faces each (cf. Figure 15 right). Note that $\frac{1}{3}c$ could be used as well, but additional steps would need to be undertaken to merge the outer quads with coinciding edges, vertices, and T-joints.

Condition B. is taken care of by checking if any S_i with i outside the 3-ring of any extraordinary vertex v overlaps the 1-ring quads of v. If this is the case, we extend the 2-ring edge chains beyond the 3-ring as depicted in the inset (red) until they intersect the knot vector determination rays (cf. Section 4.1.2) of S_i (dashed). This alters the knot intervals associated with S_i, effectively *blocking* it from reaching the 1-ring (blue). It is due to the parametric convexity of each T-mesh face (implied by four-sidedness) that at least one of the two crossed knot rays already intersects some edge somewhere within the shaded region. The intersection with the red extension is thus at least the second, meaning that the support S_i ends no further than at this point, thus not reaching the 1-ring. Note that the support of vertices lying in the white regions above or right of the 3-ring can never reach the 1-ring due to the 3-ring and 2-ring.



9 RESULTS

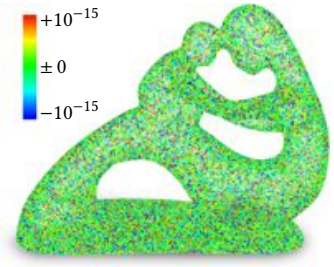
9.1 Parametrization

As a stress test, we ran our conformal parametrization algorithm on all the 88 closed models and cross fields with admissible holonomy signature from the data set published by Myles et al. [2014]. Due to containing numerous models with bad triangulation quality (slivers, needles) as well as spikes, topological noise, etc., causing points of very low or high concentrated curvature, this data set is very challenging for parametrization methods. For instance, [Chien et al. 2016] report that, while good success rates can be achieved in the larger space of non-conformal parametrizations, a classical conformal parametrization approach (*without* triangulation modification) fails on over 80% of these models. They also noted that some exemplars do not admit *any* piecewise linear parametrization (whether conformal or not) unless the triangulation can be adjusted.

Our global conformal parametrization method, with control over global map signature and integrated adaptive triangulation modification, succeeded in all cases. A selection of exemplars is shown in Figure 16. Note that these are the direct results of our conformal parametrization method; postprocessing such as described in Section 7.4 was not applied here (nor in Figures 1, 17, or 18). The input signatures' numbers k ranged from -8 to $+10$ on this data set.

Fig. 16. Examples of conformal seamless similarity parametrizations, visualized with a checkerboard texture, as computed by our method. The cuts (across which the parameters are related by seamless similarity transformations) are highlighted in yellow.

Statistics are reported in Table 1. The most important aspect is the residual, the max. and avg. curvature error, i.e. the deviation from prescribed signature, in terms of $||b(G')||$ (i.e. $|K_v - K_G[v]|$ and $|k_{\gamma_i}\frac{\pi}{2} - \kappa_g^{tot}[\gamma_i]|$ over all vertices v and homology loops i) (cf. Section 7.2). We cannot expect this residual to become exactly zero; a perfectly planar mesh (e.g. lying in the $z = 0$ plane) typically already has a curvature in the order of 10^{-15} (if the mesh is very nice and uniform) or even 10^{-10} for less nice triangulations, if computed with double precision floating point arithmetic. This is due to the limited precision when computing angles (curvature) from lengths or coordinates. The quality of the result is thus to be interpreted relative to this. As can be seen, we arrive at a max. residual of 10^{-12} or less, for the depicted cases as well as for every other case of the test set. The inset shows a visualization of the curvature error residual per vertex in the range $(-10^{-15}, +10^{-15})$. What can be observed is that the error indeed appears to be noise, rather than systematic in some form.



While the robust global seamless parametrization approach of [Myles et al. 2014] reportedly has to insert additional singularities not prescribed in the input signature in order to succeed in a few cases, our algorithms always respects the input; Figure 17 shows two of the pertinent models.

Another challenging test case is shown in Figure 18. Our algorithm successively performs 698 edge flips, and a seamless similarity parametrization is successfully obtained 7 iterations thereafter (avg. residual: 1.6×10^{-15}, max. residual: 6.3×10^{-13}, time:12min).

We also conducted tests with randomly generated input signatures, with a hundred singularities of randomly chosen extraordinary valence 2, 3, 5, or 6 distributed randomly, without any regard

Fig. 17. Models on which the method of [Myles et al. 2014] is not able to respect the input signature. Shown are seamless similarity parametrizations and exemplary derived coarse T-meshes (not optimized for alignment or refined for any particular purpose) constructed by our method, with exactly the prescribed extraordinary vertices.

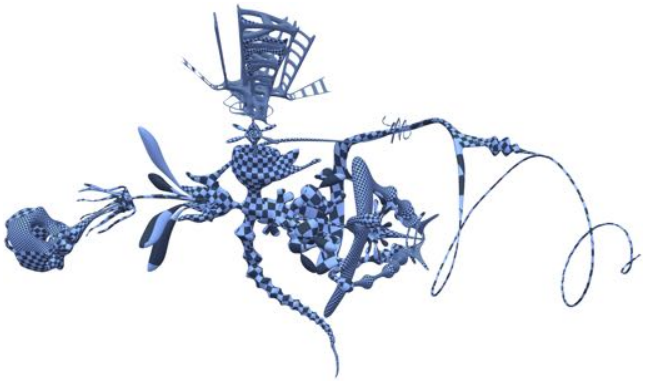


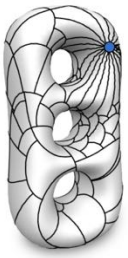
Fig. 18. Visualization of resulting seamless similarity map on a challenging model (courtesy of K. Crane): the mesh has edge lengths spanning 4 orders of magnitude, face angles ranging from 0.08° to 179.76°, face aspect ratios up to 1:1073, genus 131, and 1777 singularities in the prescribed cross field.

model	$\|F\|$	max. error	avg. error	edgeflips	w/ iDT
(1)	50K	6×10^{-14}	8×10^{-16}	14	1
(2)	100K	2×10^{-14}	7×10^{-16}	1	1
(3)	100K	9×10^{-15}	7×10^{-16}	20	3
(4)	129K	2×10^{-14}	7×10^{-16}	2	2
(5)	100K	4×10^{-14}	1×10^{-15}	7	1
(6)	100K	5×10^{-14}	8×10^{-16}	16	3
(7)	100K	9×10^{-15}	7×10^{-16}	2	1
(8)	82K	6×10^{-15}	6×10^{-16}	0	0
(9)	50K	3×10^{-14}	8×10^{-16}	14	1
(10)	100K	2×10^{-14}	8×10^{-16}	5	1
(11)	20K	5×10^{-13}	3×10^{-15}	180	2
(12)	105K	5×10^{-13}	8×10^{-15}	18	1
(13)	100K	3×10^{-13}	1×10^{-15}	75	2
(14)	18K	2×10^{-14}	9×10^{-16}	26	2
(15)	10K	3×10^{-13}	1×10^{-15}	16	2
(16)	120K	5×10^{-14}	7×10^{-16}	13	1

Table 1. Statistics of similarity map computation. Models (1)-(13) are shown in Figure 16 left to right and top to bottom, (14)-(15) in Figure 17, and model (16) in Figure 1. The average and maximum of the curvature error that remains after seven full iterations (after edge flips are done) is stated. We found that further iterations do not typically lead to further improvements. The number of edge flips executed to prevent degeneracies is also stated. In the last column we show the (typically much smaller) number of degeneracy events that need to be handled if we keep the triangulation intrinsically Delaunay at each step (cf. Section 7.3). Note that every single edge flip incurs another iteration, i.e. another linear system solve. One iteration on a model with 100K faces takes about 1 second.

to surface geometry. This is to demonstrate that success does not hinge upon benign input (e.g. signatures derived from smooth cross fields). The results are hard to visualize because the area distortion factor often spans many orders of magnitude; Figure 19 shows some of the clearer examples.

In the search for cases that numerically break the method, we found that prescribing singularities of extraordinary valence around 30 or higher (e.g. just a single singularity on a genus 4+ surface) commonly leads to higher residuals, thus to the cycle conditions being fulfilled with insufficient precision. This sometimes leads to violations of local map injectivity during metric integration or makes a derived T-mesh inadmissible. Shown here is an admissible T-mesh with a single extraordinary vertex of valence 20 successfully constructed on a genus 3 surface. Already in this case the practical relevance of such a signature prescription in our context is questionable, though there may be other use cases for such parametrizations.



9.2 T-Meshes, T-Splines, T-NURCCS

To demonstrate the validity of the smooth surface construction based on seamless similarity parametrization, we generate T-meshes from such parametrizations, computed by our method for a given extraordinary vertex distribution (in form of a cross field). Halfedge knot intervals are assigned to these as described in Section 8. Then the thus defined T-spline or T-NURCC surface can be evaluated. Figures 21 and 20 show example T-meshes and resulting surfaces, for the case of T-splines (without extraordinary vertices) and T-NURCCS (with extraordinary vertices). Figures 1, 5, and 4 show further surfaces constructed this way.

We made use of simple greedy refinement of the initial coarse T-meshes to a given uniform resolution (using 1-to-2 face splits), and of least-squares fitting [Greiner 1994; Li et al. 2006] to determine appropriate positions for the control vertices. Note that the main purpose here is to demonstrate the proper functioning of our approach; we do not make novel contributions in the fitting or control mesh adaptation regime. Depending on the application scenario, different fitting and refinement strategies will be preferable.

10 LIMITATIONS AND FUTURE WORK

Limitations. An inherent property of the proposed method is that it invariably produces meshes with T-joints even in cases when conformal meshes do exist. This is due to the fact that we fix the scale factors after computing the conformal map, and these are uniquely determined by the choice of rotations on loops. These factors determine how many T-joints may be needed to obtain a T-mesh with a sufficiently uniform size of patches on the surface. It is important to notice, though, that T-joints, much unlike extraordinary vertices,

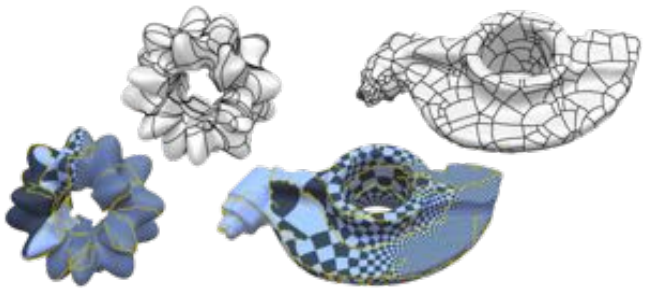


Fig. 19. Parametrizations obtained for randomly generated singularity distributions. The scale jumps across cuts can be very large in such cases, yet the maps are valid, thus, e.g., suitable to define admissible T-meshes as shown on the top (without regular refinement around extraordinary vertices for clarity).

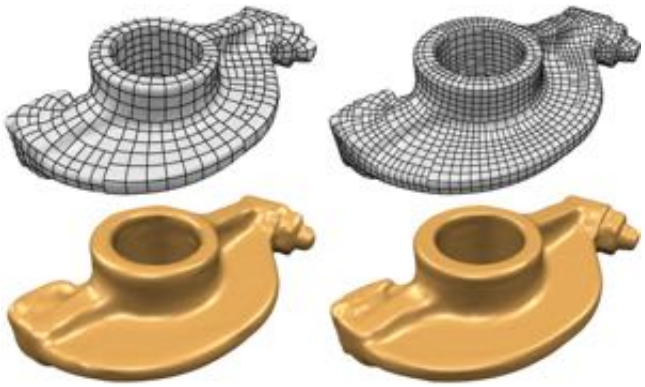


Fig. 21. T-mesh and T-spline surface with no extraordinary vertices on a genus 1 model, at two different control mesh resolutions.

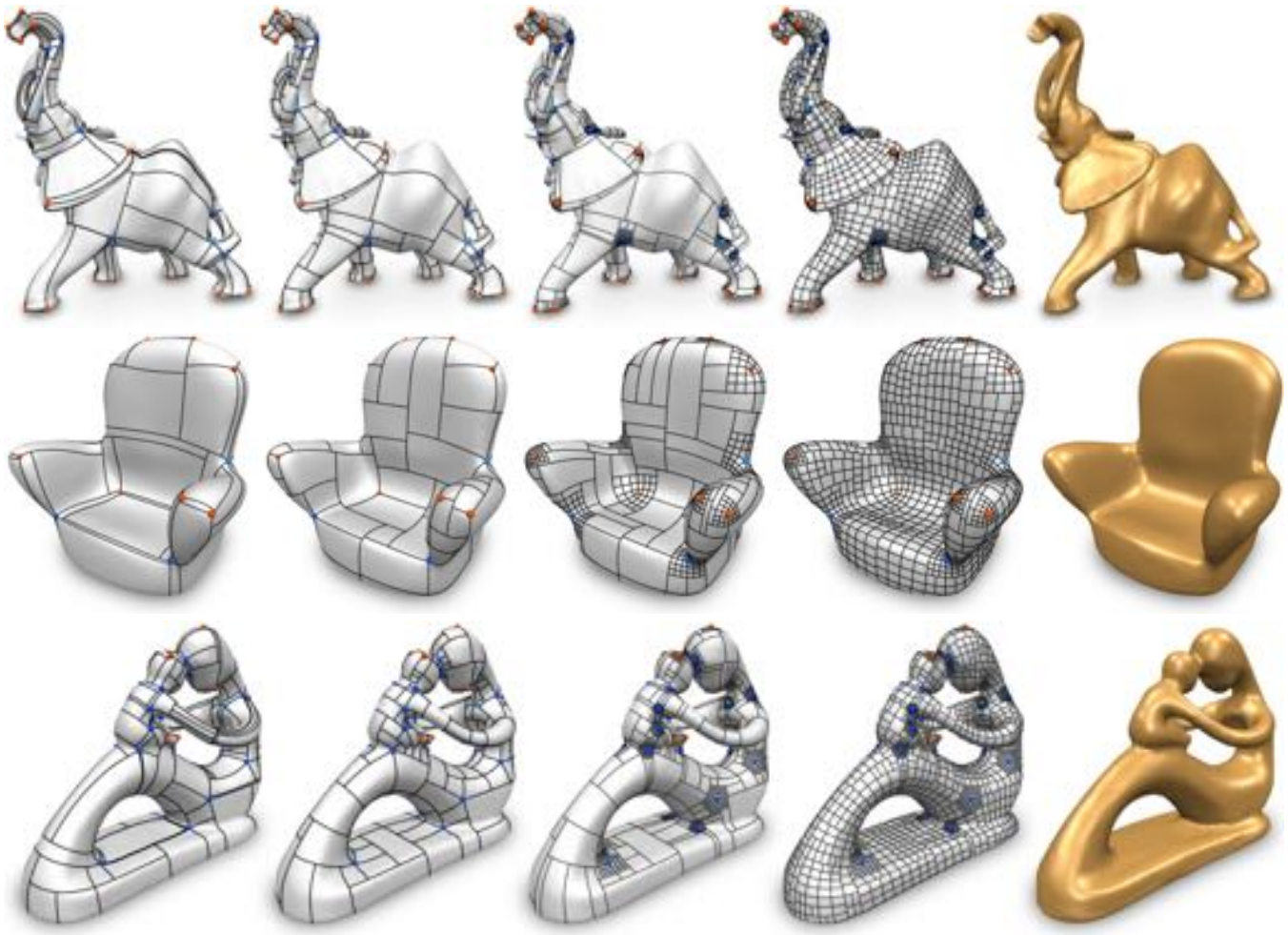


Fig. 20. T-meshes extracted from seamless similarity parametrizations. From left to right: 1) the initial motorcycle graph, 2) the regularized version, 3) regular layers inserted around extraordinary vertices, 4) a uniform refinement to a desired resolution, and 5) a rendering of the fitted T-spline or T-NURCC surface.

do not deteriorate surface quality in terms of continuity, but there might still be reasons to prefer meshes with mostly regular joints.

So far, there is no complete proof guaranteeing convergence for our discrete global conformal parametrization algorithm; particularly the potentially adverse effects of triangulation modification are not yet fully understood [Campen and Zorin 2017].

Future work. The novel possibilities provided by a) our discrete conformal parametrization algorithm with global holonomy control, and b) our extended T-spline definition with halfedge knots, open up a number of interesting avenues for future investigation.

We have made use of similarity transitions in our work; these involve *isotropic* scaling. In theory, one can imagine the use of transitions that scale anisotropically, i.e. the two parametric dimensions (u and v) differently. How domains with such transitions can be constructed algorithmically, and where this increased flexibility is of benefit needs to be investigated.

The scale factors determined by the conformal maps can be optimized in a variety of ways, most importantly, driving them to 1 whenever possible to create conforming meshes with no T-joints, or powers-of-two, to create dyadic T-meshes.

Conceptually, most of our considerations easily extend to surfaces with boundary or with sharp features (note that sharp feature curves can be treated as interpolated boundaries for the purpose of parametrization, mesh constructions, and evaluation). There are a number of details though (e.g. related to the triangulation modification or the T-mesh regularization) which need algorithmic attention.

We have seen that T-meshes with a variety of structural properties can be derived. We consider important looking into further ways to define T-meshes of high quality based on a seamless similarity parametrization, and investigating what actually makes a high-quality control T-mesh for specific application scenarios.

ACKNOWLEDGMENTS

The authors thank Justin Bisceglio for his support with implementing the surface evaluation, especially around extraordinary vertices. Models are courtesy of Carlo Séquin and the Aim@Shape repository.

REFERENCES

Ralph Abraham, Jerrold E Marsden, and Tudor Ratiu. 1988. *Manifolds, Tensor Analysis, and Applications.* Springer, New York, NY, USA.

Noam Aigerman and Yaron Lipman. 2015. Orbifold Tutte Embeddings. *ACM Trans. Graph.* 34, 6 (2015), 190:1–190:12.

Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. 2003. Anisotropic polygonal remeshing. *ACM Trans. Graph.* 22, 3 (2003).

Thierry Aubin. 1998. *Some Nonlinear Problems in Riemannian Geometry.* Springer Berlin Heidelberg.

Yuri Bazilevs, Victor M Calo, John A Cottrell, John A Evans, TJR Hughes, S Lipton, Michael A Scott, and Thomas W Sederberg. 2010. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Eng.* 199, 5 (2010), 229–263.

Mirela Ben-Chen, Craig Gotsman, and Guy Bunin. 2008. Conformal flattening by curvature prescription and metric scaling. In *Comp. Graph. Forum*, Vol. 27. 449–458.

Martin Bertram, Mark A. Duchaineau, Bernd Hamann, and Kenneth I. Joy. 2004. Generalized B-Spline Subdivision-Surface Wavelets for Geometry Compression. *IEEE TVCG* 10, 3 (2004), 326–338.

Ioana Boier-Martin, Holly Rushmeier, and Jingyi Jin. 2004. Parameterization of triangle meshes over quadrilateral domains. *Symp. on Geometry Processing* (2004), 193–203.

David Bommes, Marcel Campen, Hans-Christian Ebke, Pierre Alliez, and Leif Kobbelt. 2013. Integer-grid maps for reliable quad meshing. *ACM Trans. Graph.* 32, 4 (2013).

David Bommes, Henrik Zimmer, and Leif Kobbelt. 2009. Mixed-integer quadrangulation. *ACM Trans. Graph.* 28, 3 (2009), 77.

Philip L. Bowers and Monica K. Hurdal. 2003. *Planar Conformal Mappings of Piecewise Flat Surfaces.* Springer Berlin Heidelberg, 3–34.

Guy Bunin. 2008. A continuum theory for unstructured mesh generation in two dimensions. *Computer Aided Geometric Design* 25, 1 (2008), 14–40.

Marcel Campen, David Bommes, and Leif Kobbelt. 2015. Quantized global parametrization. *ACM Trans. Graph.* 34, 6 (2015), 192.

Marcel Campen and Leif Kobbelt. 2014. Quad Layout Embedding via Aligned Parameterization. *Comp. Graph. Forum* 33, 8 (2014).

Marcel Campen and Denis Zorin. 2017. On Discrete Conformal Seamless Similarity Maps. (2017). arXiv:1705.02422

Nathan A. Carr, Jared Hoberock, Keenan Crane, and John C. Hart. 2006. Rectangular multi-chart geometry images. In *Proc. SGP '06.* 181–190.

Thomas J. Cashman. 2012. Beyond Catmull-Clark? A Survey of Advances in Subdivision Surface Methods. *Comp. Graph. Forum* 31, 1 (2012), 42–61.

Edward Chien, Zohar Levi, and Ofir Weber. 2016. Bounded Distortion Parametrization in the Space of Metrics. *ACM Trans. Graph.* 35, 6 (2016), 215:1–215:16.

Bennett Chow and Feng Luo. 2003. Combinatorial Ricci Flows on Surfaces. *J. Differential Geom.* 63, 1 (2003), 97–129.

Keenan Crane, Mathieu Desbrun, and Peter Schröder. 2010. Trivial Connections on Discrete Surfaces. *Comp. Graph. Forum* 29, 5 (2010), 1525–1533.

Jiansong Deng, Falai Chen, Xin Li, Changqi Hu, Weihua Tong, Zhouwang Yang, and Yuyu Feng. 2008. Polynomial Splines over Hierarchical T-meshes. *Graph. Models* 70, 4 (2008), 76–86.

Mathieu Desbrun, Eva Kanso, and Yiying Tong. 2008. *Discrete Differential Forms for Computational Modeling.* Birkhäuser Basel, 287–324.

Hans-Christian Ebke, Patrick Schmidt, Marcel Campen, and Leif Kobbelt. 2016. Interactively Controlled Quad Remeshing of High Resolution 3D Models. *ACM Trans. Graph.* 35, 6 (2016), 218:1–218:13.

Matthias Eck and Hugues Hoppe. 1996. Automatic Reconstruction of B-spline Surfaces of Arbitrary Topological Type. In *Proc. (SIGGRAPH '96).* 325–334.

David Eppstein, Michael T. Goodrich, Ethan Kim, and Rasmus Tamstorf. 2008. Motorcycle Graphs: Canonical Quad Mesh Partitioning. *Comp. Graph. Forum* 27, 5 (2008), 1477–1486.

Gerald Farin. 2002. *Curves and Surfaces for CAGD: A Practical Guide* (5th ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Matthew Fisher, Boris Springborn, Peter Schröder, and Alexander I Bobenko. 2007. An algorithm for the construction of intrinsic Delaunay triangulations with applications to digital geometry processing. *Computing* 81, 2-3 (2007), 199–213.

Gunther Greiner. 1994. Variational Design and Fairing of Spline Surfaces. *Comp. Graph. Forum* 13, 3 (1994).

Xianfeng Gu, Ying He, Miao Jin, Feng Luo, Hong Qin, and Shing-Tung Yau. 2008. Manifold splines with a single extraordinary point. *Computer-Aided Design* 40, 6 (2008), 676–690.

Xianfeng Gu, Feng Luo, Jian Sun, and Tianqi Wu. 2013. A discrete uniformization theorem for polyhedral surfaces. *arXiv:1309.4175* (2013).

Xianfeng Gu and Shing-Tung Yau. 2003. Global conformal surface parameterization. In *Proc. Symposium on Geometry Processing 2003.* 127–137.

Ying He, Kexiang Wang, Hongyu Wang, Xianfeng Gu, and Hong Qin. 2006. Manifold T-Spline. In *Proc. Geometric Modeling and Processing.* 409–422.

Miao Jin, Junho Kim, and Xianfeng David Gu. 2007. *Discrete Surface Ricci Flow: Theory and Applications.* 209–232.

Ernest Jucovič and Marián Trenkler. 1973. A theorem on the structure of cell–decompositions of orientable 2–manifolds. *Mathematika* 20, 01 (1973), 63–82.

Felix Kälberer, Matthias Nieser, and Konrad Polthier. 2007. QuadCover: Surface Parameterization using Branched Coverings. *Comp. Graph. Forum* 26, 3 (2007), 375–384.

Kestutis Karciauskas, Daniele Panozzo, and Jörg Peters. 2016. Spline surfaces with T-junctions. (2016). arXiv:1610.05351.

Liliya Kharevych, Boris Springborn, and Peter Schröder. 2006. Discrete Conformal Mappings via Circle Patterns. *ACM Trans. Graph.* 25, 2 (2006), 412–438.

Andrei Khodakovsky, Peter Schröder, and Wim Sweldens. 2000. Progressive Geometry Compression. In *Proc. SIGGRAPH '00.* 271–278.

Denis Kovacs, Justin Bisceglio, and Denis Zorin. 2015. Dyadic T-mesh Subdivision. *ACM Trans. Graph.* 34, 4 (2015), 143:1–143:12.

Wan-Chiu Li, Nicolas Ray, and Bruno Lévy. 2006. Automatic and interactive mesh to T-spline conversion. In *Proc. SGP '06.* 191–200.

Xin Li, Jiansong Deng, and Falai Chen. 2010. Polynomial splines over general T-meshes. *The Visual Computer* 26, 4 (2010), 277–286.

Xin Li, G. Thomas Finnigan, and Thomas W. Sederberg. 2016. G1 Non-uniform Catmull-Clark Surfaces. *ACM Trans. Graph.* 35, 4 (2016), 135:1–135:8.

Xin Li, Jianmin Zheng, Thomas W. Sederberg, Thomas J. R. Hughes, and Michael A. Scott. 2012. On linear independence of T-spline blending functions. *Computer Aided Geometric Design* 29, 1 (2012), 63–76.

Yaron Lipman. 2012. Bounded distortion mapping spaces for triangular meshes. *ACM Trans. Graph.* 31, 4 (2012), 108.

Michael Lounsbery, Tony D. DeRose, and Joe Warren. 1997. Multiresolution Analysis for Surfaces of Arbitrary Topological Type. *ACM Trans. Graph.* 16, 1 (1997), 34–73.

Feng Luo. 2004. Combinatorial Yamabe flow on surfaces. *Communications in Contemporary Mathematics* 6, 05 (2004), 765–780.

Kerstin Müller, Lars Reusche, and Dieter Fellner. 2006. Extended Subdivision Surfaces: Building a Bridge Between NURBS and Catmull-Clark Surfaces. *ACM Trans. Graph.* 25, 2 (2006), 268–292.

Ashish Myles, Nico Pietroni, Denis Kovacs, and Denis Zorin. 2010. Feature-aligned T-meshes. *ACM Trans. Graph.* 29, 4 (2010), 117:1–117:11.

Ashish Myles, Nico Pietroni, and Denis Zorin. 2014. Robust Field-aligned Global Parametrization. *ACM Trans. Graph.* 33, 4 (2014).

Ashish Myles and Denis Zorin. 2012. Global parametrization by incremental flattening. *ACM Trans. Graph.* 31, 4 (2012), 109.

Ashish Myles and Denis Zorin. 2013. Controlled-distortion constrained global parametrization. *ACM Trans. Graph.* 32, 4 (2013), 105.

Daniele Panozzo, Enrico Puppo, Marco Tarini, Nico Pietroni, and Paolo Cignoni. 2011. Automatic Construction of Quad-Based Subdivision Surfaces Using Fitmaps. *IEEE Transactions on Visualization and Computer Graphics* 17, 10 (2011), 1510–1520.

Jörg Peters. 1992. Joining smooth patches around a vertex to form a C^k surface. *Computer Aided Geometric Design* 9, 5 (1992), 387 – 411.

M. M. Postnikov. 2001. *Geometry VI: Riemannian Geometry.* Springer Berlin Heidelberg.

Nicolas Ray, Wan Chiu Li, Bruno Lévy, Alla Sheffer, and Pierre Alliez. 2006. Periodic Global Parameterization. *ACM Trans. Graph.* 25, 4 (2006), 1460–1485.

Nicolas Ray, Vincent Nivoliers, Sylvain Lefebvre, and Bruno Lévy. 2010. Invisible Seams. In *Proc. Eurographics Symposium on Rendering.*

Nicolas Ray and Dmitry Sokolov. 2014. Robust Polylines Tracing for N-Symmetry Direction Field on Triangulated Surfaces. *ACM Trans. Graph.* 33, 3 (2014).

Nicolas Ray, Bruno Vallet, Wan Chiu Li, and Bruno Lévy. 2008. N-Symmetry direction field design. *ACM Trans. Graph.* 27 (2008), 2.

Richard Schoen and Shing-Tung Yau. 1994. *Lectures on Differential Geometry.* International Press.

Thomas W. Sederberg, David L. Cardon, G. Thomas Finnigan, Nicholas S. North, Jianmin Zheng, and Tom Lyche. 2004. T-spline Simplification and Local Refinement. *ACM Trans. Graph.* 23, 3 (2004), 276–283.

Thomas W. Sederberg, G. Thomas Finnigan, Xin Li, Hongwei Lin, and Heather Ipson. 2008. Watertight trimmed NURBS. *ACM Trans. Graph.* 27, 3 (2008).

Thomas W. Sederberg, Jianmin Zheng, Almaz Bakenov, and Ahmad Nasri. 2003. T-splines and T-NURCCs. *ACM Trans. Graph.* 22, 3 (2003), 477–484.

Boris Springborn, Peter Schröder, and Ulrich Pinkall. 2008. Conformal equivalence of triangle meshes. *ACM Trans. Graph.* 27, 3 (2008).

Jos Stam. 1998. Exact Evaluation of Catmull-Clark Subdivision Surfaces at Arbitrary Parameter Values. In *Proc (SIGGRAPH '98).* 395–404.

Yiying Tong, Pierre Alliez, Daniel Cohen-Steiner, and Mathieu Desbrun. 2006. Designing quadrangulations with discrete harmonic forms. *Symposium on Geometry Processing* (2006), 201–210.

Marc Troyanov. 1991. Prescribing curvature on compact surfaces with conical singularities. *Trans. Amer. Math. Soc.* 324 (1991), 793–821.

Amir Vaxman, Marcel Campen, Olga Diamanti, Daniele Panozzo, David Bommes, Klaus Hildebrandt, and Mirela Ben-Chen. 2016. Directional Field Synthesis, Design, and Processing. *Comp. Graph. Forum* 35, 2 (2016).

Wenyan Wang, Yongjie Zhang, Guoliang Xu, and ThomasJ.R. Hughes. 2012. Converting an unstructured quadrilateral/hexahedral mesh to a rational T-spline. *Computational Mechanics* 50, 1 (2012), 65–84.

Denis Zorin, Peter Schröder, and Wim Sweldens. 1997. Interactive Multiresolution Mesh Editing. In *Proc. SIGGRAPH '97.* 259–268.

A PROOFS

Proof of Proposition 1

Proof. The invariance of $\mathcal{F}$ to seamless similarity transforms applied to domains and knot vectors is a direct consequence of the invariance of each basis function to such transforms, as established by Proposition 2. As knot vectors are determined with respect to the local systems in D, and these local systems differ between different choices of anchor quads by seamless similarity transforms only (namely compositions of $\tau[F_m, F_n]$ and translations, all of which are seamless similarity transforms), $\mathcal{F}$ is in particular invariant to the choice of anchor quads and local coordinate systems. As a special case, this implies that the quadrant basis functions of one basis functions, even if defined with respect to individual anchors, thus different local systems, join in a C^2 (or "G^2") manner, just as if they were defined with respect to the same system, thus were simply restrictions of the same function.

Continuity order C^2 of $\mathcal{F}$ away from extraordinary points follows from the C^2 continuity of each basis function (composed of quadrants) and all applied transforms $\tau[F, D]$ being smooth similarity transformations. $\mathcal{F}$ is piecewise rational, away from extraordinary points, because the underlying construction of T-splines (1) yields piecewise rational functions [Sederberg et al. 2003]. □

Proof of Proposition 2

Proof. Indeed, nontrivial rotations by $k\pi/2$ are a combination of flipping the signs of knots with possibly switching u and v directions, with $u_i = \pm s u_i + t_u$ or $u_i = \pm s v_i + t_u$. Consider one of the one-dimensional factors $B[\mathbf{u}](u)$ in the formula. Clearly, for a spline of order 0, the proposition holds. Next, we substitute the transform formulas into the recursive definition:

$$B_{i,k}(u) = \frac{u - u_i}{u_{i+k-1} - u_i} B_{i,k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_{i+1}} B_{i+1,k-1}(u),$$

where the i-th basis function is supported on $u_i, \ldots, u_{i+k}$.

If u and v are not switched, we observe that the coefficients of the formula are invariant with respect to the transformation: $\frac{u' - u'_i}{u'_{i+k-1} - u'_i} = \frac{u - u_i}{u_{i+k-1} - u_i}$, and we conclude that $B[\mathbf{u}'](u') = B(\mathbf{u})(u)$ by induction on order. The same holds for $B[\mathbf{v}]$. If u and v are switched by the rotation, in a similar way we observe that $B[\mathbf{u}'](u') = B[\mathbf{v}](v)$ and $B[\mathbf{v}'](v') = B[\mathbf{u}](u)$. As $B[\mathbf{u}', \mathbf{v}'](\mathbf{u}', \mathbf{v}') = B[\mathbf{u}'](u') B[\mathbf{v}'](v')$, this yields the formula in the proposition. □

Proof of Proposition 4

Proof. It remains to show that the conformal map f is a seamless similarity map, i.e. the transition functions across cuts are similarities that are constant on each branch (segment not containing a singularity or branching point) of the cut graph.

As $\xi = \star\omega_f$, defined on M, is continuous on the cut, we have $d\phi_1^c = \xi = d\phi_2^c$, where ϕ_i^c are the two values of ϕ on different sides of the cut. Integrating along the cut, we see that the changes of ϕ_1^c and ϕ_2^c between two points $\mathbf{p}$ and $\mathbf{q}$ on a cut branch are the same, i.e. $\phi_1^c(\mathbf{q}) - \phi_1^c(\mathbf{p}) = \phi_2^c(\mathbf{q}) - \phi_2^c(\mathbf{p})$. From this we conclude that $\phi_1^c(\mathbf{q}) - \phi_2^c(\mathbf{q}) = \phi_1^c(\mathbf{p}) - \phi_2^c(\mathbf{p})$, i.e. jumps in ϕ^c at different points $\mathbf{p}$ and $\mathbf{q}$ on a branch are the same, i.e. the ratio of the scale factors $e^{\phi_1^c} / e^{\phi_2^c}$ is constant along a branch.

Concerning the rotational part, consider $Df_1[\mathbf{p}]$ and $Df_2[\mathbf{p}]$, the differentials of the map f on the two different sides of the cut at a point $\mathbf{p}$ on the cut, and a loop γ that crosses the cut once at $\mathbf{p}$. As ω_f measures the rate of rotation of the cross field defined by $Df^{-1}(\mathbf{e}_1)$, and as its integral over any loop γ cancels its geodesic curvature up to a multiple of $\pi/2$ by construction of f, the rotation between $Df_2^{-1}[\mathbf{p}](\mathbf{e}_1)$ and $Df_1^{-1}[\mathbf{p}](\mathbf{e}_1)$ is a multiple of $\pi/2$, i.e. $Df_2^{-1}[\mathbf{p}](\mathbf{e}_1) = sRDf_1^{-1}[\mathbf{p}](\mathbf{e}_1)$, where R is a rotation by a multiple of $\pi/2$. Note that Df (and thus Df^{-1}) is a rotation + uniform scale for a fixed pair of coordinate systems on the tangent plane and in the parametric domain. Defining a rotation + uniform scale transform on a single vector ($\mathbf{e}_1$) completely determines it, so we conclude that $Df_2^{-1}[\mathbf{p}] = sRDf_1^{-1}[\mathbf{p}]$. As two-dimensional rotations commute, we obtain $Df_2^{-1}[\mathbf{p}] Df_1[\mathbf{p}] = sR$, i.e. the differential of the transition map from one side of the cut to the other is a $k\pi/2$ rotation composed with a uniform scale. As the rotation along the loop depends on the regular homotopy class of the loop only, at all points of the cut branch, the rotation and thus the differential of the map is the same.

Finally, concerning the translational part, integrating differentials from a point on a cut branch to any other point along the branch by the same argument used for ϕ, we arrive at the conclusion that the transitions per cut branch are constant similarity maps. □

B INDICATOR FUNCTION

Consider the function $I'_{ij,k}(\lambda) =$

$$\ell_{jk} e^{(\phi_j(\lambda\xi) + \phi_k(\lambda\xi))/2} + \ell_{ki} e^{(\phi_k(\lambda\xi) + \phi_i(\lambda\xi))/2} - \ell_{ij} e^{(\phi_i(\lambda\xi) + \phi_j(\lambda\xi))/2}.$$

It is a sum of (negated) edge lengths at time λ of the metric evolution. It is negative if the triangle inequality of triangle ijk is violated by edge ij being too long, vanishes if edge ij is as long as the other two edges combined, and positive otherwise. It thus has the desired degeneracy-indicating behavior. Division by the strictly positive (thus root preserving) term $e^{(\phi_i(\lambda\xi) + \phi_j(\lambda\xi))/2}$ yields (9):

$$\ell_{jk} e^{(\phi_k(\xi) - \phi_i(\xi))\lambda/2} + \ell_{ki} e^{(\phi_k(\xi) - \phi_j(\xi))\lambda/2} - \ell_{ij} = I_{ij,k}(\lambda),$$

using $\phi_i(\xi) - \phi_j(\xi) = \xi_{ji}$ for any two adjacent vertices i, j. Notice that $I_{ij,k}$ can be evaluated locally directly based on ξ; global integration to obtain ϕ is not required.

Furthermore, the second derivative of (9) is strictly positive

$$\frac{d^2}{d\lambda^2} I_{ij,k} = \ell_{jk} \tfrac{1}{4} \xi_{ik}^2 e^{\xi_{ik}\lambda/2} + \ell_{ki} \tfrac{1}{4} \xi_{jk}^2 e^{\xi_{jk}\lambda/2} > 0,$$

assuming that $\xi_{ik} \neq 0$ or $\xi_{jk} \neq 0$ (otherwise the corresponding triangle remains constant and does not need to be checked for degeneracy events anyway). This is due to ℓ_{jk} and ℓ_{ki} being positive by construction. Thus I has at most two roots.