Seamless Parametrization with Arbitrary Cones for Arbitrary Genus

MARCEL CAMPEN, Osnabrück University, Germany
HANXIAO SHEN, New York University, USA
JIARAN ZHOU, Shandong University, China
DENIS ZORIN, New York University, USA

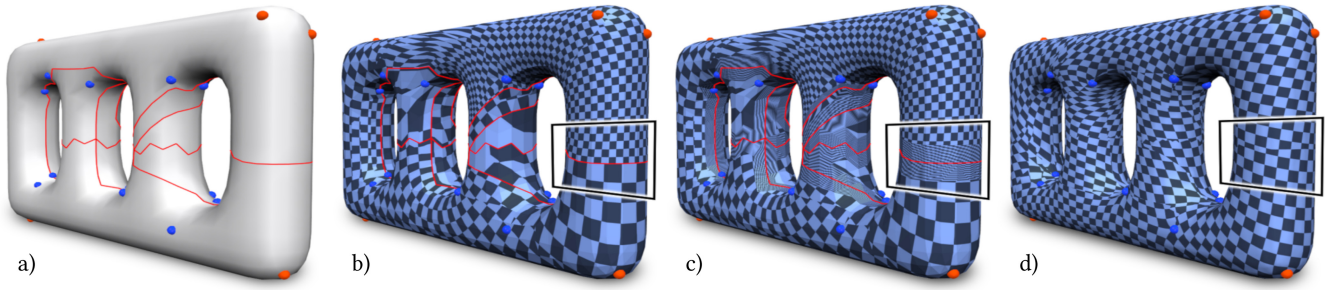


Fig. 1. Method overview: (a) Cut graph on a surface, consisting of handle loops, connectors, and one additional path. (b) Conformal parametrization that maps the cut graph's branches to axis-aligned straight segments in the parametric domain and respects prescribed cone singularities (red and blue dots). This map is only *rotationally* seamless, i.e., rotational components of transitions across cuts are $k\pi/2$-rotations, $k \in \mathbb{Z}$, but scaling is arbitrary. (c) This map modified by *map padding*; while locally highly distorted, it is actually seamless, there no longer is a scale jump. (d) Result after optimization for low isometric distortion.

Seamless global parametrization of surfaces is a key operation in geometry processing, e.g., for high-quality quad mesh generation. A common approach is to prescribe the parametric domain structure, in particular, the locations of parametrization singularities (cones), and solve a non-convex optimization problem minimizing a distortion measure, with local injectivity imposed through either constraints or barrier terms. In both cases, an initial valid parametrization is essential to serve as a feasible starting point for obtaining an optimized solution. While convexified versions of the constraints eliminate this initialization requirement, they narrow the range of solutions, causing some problem instances that actually do have a solution to become infeasible.

We demonstrate that for arbitrary given sets of topologically admissible parametric cones with prescribed curvature, a global seamless parametrization always exists (with the exception of one well-known case). Importantly, our proof is constructive and directly leads to a general algorithm for computing such parametrizations. Most distinctively, this algorithm is bootstrapped with a convex optimization problem (solving for a conformal map), in tandem with a simple linear equation system (determining a seamless modification of this map). This initial map can then serve as a valid starting point and be optimized for low distortion using existing injectivity preserving methods.

CCS Concepts: • **Computing methodologies** → **Mesh geometry models**; **Parametric curve and surface models**;

Additional Key Words and Phrases: Conformal, cone metric, holonomy

ACM Reference format:
Marcel Campen, Hanxiao Shen, Jiaran Zhou, and Denis Zorin. 2019. Seamless Parametrization with Arbitrary Cones for Arbitrary Genus. *ACM Trans. Graph.* 39, 1, Article 2 (December 2019), 19 pages.
https://doi.org/10.1145/3360511

1 INTRODUCTION

Computing global parametrizations of surfaces is a key operation in geometry processing. While in general only disk-like surfaces can be parametrized continuously in a (locally or globally) injective manner, surfaces of arbitrary topology can be dealt with by cutting them to disks. Across the cuts, the parametrization will be discontinuous, but this is inevitable, in general.

One can, however, require the parametric transitions across cuts to be from certain classes, to support specific applications like smooth surface approximation and quadrangulation. For instance, restricting to *similarity* transformations (rotation, translation, isotropic scaling) with a rotation by some multiple of $\pi/2$ yields global parametrizations ideal for T-spline constructions [Campen and Zorin 2017]. Restricting further to *rigid* transformations with such discrete rotation angles yields parametrizations that (after quantization [Bommes et al. 2013a; Campen et al. 2015; Lyon et al. 2019]) are well suited for tasks like conforming quadrangulation, spline and subdivision fitting, seamless texturing, or constructing grids for solving PDEs on surfaces. We call such parametrizations *seamless* [Myles and Zorin 2012; Purnomo et al. 2004].

H. Shen and D. Zorin are supported by Awards No. NSF IIS-1320635, No. NSF DMS-1436591, and a gift from Adobe.
Authors' addresses: M. Campen, Institut für Informatik, Universität Osnabrück, Osnabrück, Germany; email: campen@uos.de; H. Shen, Computer Science Department, New York University, New York, USA; email: hs3048@nyu.edu; J. Zhou, School of Computer Science and Technology, Shandong University, Jinan, China; email: jiaran.zhou@gmail.com; D. Zorin, Computer Science Department, New York University, New York, USA; email: dzorin@cs.nyu.edu.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
0730-0301/2019/12-ART2 $15.00
https://doi.org/10.1145/3360511

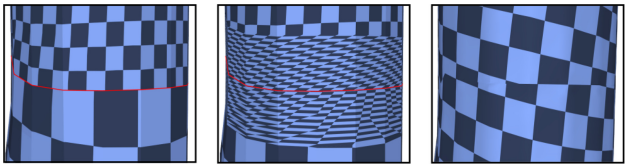


Fig. 2. Zoom-ins of Figure 1. Left: cut-aligned conformal map. Middle: padded map, with high distortion, but seamless and locally injective. Right: map optimized for low isometric distortion.

Seamless parametrizations can have *singularities*, points around which the total parametric angle is not 2π but some other integer multiple of $\pi/2$, i.e., the parametrization coordinate isolines do not locally form a regular grid. Equivalently, the metric induced by the parametrization has *cones*, points where the metric is not flat, its curvature not zero but some other integer multiple of $\pi/2$. Intuitively, in a quadrangulation induced by the parametrization, these singularities or cones correspond to extraordinary vertices, with valence different from 4.

As implied by the Gauss-Bonnet theorem, the total curvature of these cones is a topological invariant—i.e., such cones, which have a significant influence on the parametrization's quality and structure, cannot generally be avoided. Depending on the use case, they can be considered an impairment or features of special interest. In either case, having the ability to control (i.e., prescribe) them—where they are, how many there are, what curvature they have—is of obvious benefit. This motivates the problem we consider in this article:

Compute a global seamless locally injective parametrization with prescribed cone positions and curvatures (respecting Gauss-Bonnet).

Known general approaches to this problem (cf. Section 2), e.g., those used as the initial, and most difficult, step in quadrangulation algorithms, rely on optimization formulations that are non-convex and require feasible starting points to guarantee success; alternatively, convexified formulations may not yield a solution even if one exists. Thus, formally establishing existence of a solution and constructing a feasible starting point is an essential step in the general case. For the special case of genus 0, this task can be handled using existing conformal metric computation techniques [Springborn et al. 2008; Luo 2004; Gu et al. 2018b; Campen and Zorin 2017], cf. Section 4.3. Close to a general reliable solution to this problem is an approach by Myles et al. [2014]: a valid global seamless injective parametrization is guaranteed, cone preservation is aimed for but not guaranteed—in a small fraction of cases unnecessary additional cones arise.

That these are truly unnecessary in almost all cases follows from the fact that the above task is actually feasible: The *existence* of such parametrizations follows from a theorem on meshes with prescribed extraordinary vertices [Jucovič and Trenkler 1973]. The proof is relatively complex and purely combinatorial. As a consequence, it does not easily translate into a practical parametrization construction.

In this article, we provide a *constructive* proof for the existence of seamless surface parametrizations that is conceptually simpler and translates to a parametrization algorithm. Precisely, we show:

Theorem 1.1. *Given a closed smooth surface M of genus g and an admissible set C of cones c_i, each given by a point $p_i \in M$ with a prescribed curvature value $\hat{\Theta}_i = (4 - k_i)\frac{\pi}{2}$, $k_i \in \mathbb{N}^{>1}$, there exists a global parametrization of M with cones C that has seamless transitions.*[1]

The terms used in the theorem are defined precisely in Section 3.

A set of cones $C = \{(p_i, k_i)\}$ is called *admissible* if it satisfies $\sum_i (1 - \frac{1}{4}k_i) = 2 - 2g$ (Gauss-Bonnet) and if $\boldsymbol{k} \neq (3, 5)$ (which is the single one notorious infeasible case [Jucovič and Trenkler 1973]).

Basic Idea

Instead of directly aiming for a seamless cone metric on a surface M:

(1) we cut M open using a cut graph G, obtaining the cut surface M' consisting of one or more topological disks;
(2) compute a cone metric on M'—*without* any seamlessness requirements, but with prescribed boundary curvature; specifically, we prescribe a *rectilinear* boundary, consisting of geodesically straight segments meeting at right angles;
(3) modify this metric into a seamless one on M, yielding a seamless parametrization with the prescribed cones; exploiting the rectilinear boundary property, this modification is performed by *padding* the straight segments in the parametric domain with rectangles of suitably chosen size.

The metric in step (2) is known to exist; e.g., a *conformal* metric with prescribed cones and boundary curvature (satisfying Gauss-Bonnet) on a disk always exists—in the smooth setting (cf. Section 3.2); the situation is more complicated in the discrete setting (cf. Section 5.2): questions concerning the exact conditions for existence of *discrete* conformal metrics with prescribed boundary curvature as well as concerning the convergence of the existing algorithm that we leverage for this step, thus the injectivity of the derived map, remain open. We note that conformality is not essential here: Any metric with prescribed boundary curvature could be used in that step.

Figures 1 and 2 show the outcome of these main steps. We refer to Appendix A for a comprehensive example illustrating these steps in a concrete, simple case.

Key Contributions

Our key technical contributions pertain to step (3) in the outline above:

- We propose a technique (*map padding*) to modify a non-seamless map into a seamless one.
- We prove that, for certain choices of cut graph combinatorics, this technique always succeeds.
- We describe an implementation of this construction for the discrete, piecewise linear case.

[1] We assume $k_i > 1$, as cones with curvature $3\frac{\pi}{2}$, corresponding to valence 1 vertices in a quadrangulation, are of low relevance in common applications; with some additional special case handling, our method could be extended to $k_i = 1$.

In other words, our approach is a problem reduction:

If one is able to compute a metric with prescribed cones and prescribed boundary curvature for *disk-topology surfaces*, then this solves (by means of our technique) the more general problem of computing a global seamless parameterization with prescribed cones for *arbitrary-topology surfaces*.

Our algorithm can be used to obtain non-degenerate, locally injective, seamless parametrizations of arbitrary closed discrete surfaces (triangle meshes) with arbitrary cones, assuming the initial metric with prescribed boundary curvature can be obtained.

2 RELATED WORK

Seamless surface parametrization and the related subject of quadrangulation and quad layout generation is a well-explored topic. A relatively recent survey [Bommes et al. 2013b] has references to many works in this area. We focus here on the most closely related ones.

In a wide variety of applications, surface parametrizations are required to be (locally) injective (i.e., without fold-overs) as well as to exhibit low parametric distortion [Floater and Hormann 2005]. Due to the challenging nature of this requirement, a common strategy is to proceed in a two-step fashion: First, construct an initial injective parametrization (without specific attention to distortion), then optimize it with respect to application specific distortion criteria (while preserving injectivity). Our work follows this strategy.

Constructing injective maps. Whenever a robust overall algorithm is desired, injective maps are almost always initialized using the same classical result on convex harmonic maps [Floater 1997; Tutte 1963] (essentially a discrete version of the Radó-Kneser-Choquet theorem). In its original form, it handles surfaces with disk topology and does not support cones. Some recent results [Aigerman and Lipman 2015, 2016; Bright et al. 2017; Gortler et al. 2006] elegantly generalize the idea to other settings, but either not to arbitrary sets of cones, not to arbitrary topology, not using the piecewise linear Euclidean setting, or without similar guarantees on map existence.

Injectivity-preserving optimization. A variety of techniques have been presented for distortion optimization (e.g., Hormann and Greiner [2000]; Kovalsky et al. [2016]; Rabinovich et al. [2017]; Schüller et al. [2013]; Shtengel et al. [2017]; Zhu et al. [2018]). Through line search techniques, barrier functions, and similar techniques they are able to guarantee preservation of injectivity—if initialized with an injective starting point. State-of-the-art techniques can handle large meshes efficiently and tolerate significant imperfections in the initial solution.

Seamless parametrization. A number of methods have been described for the construction of seamless parametrizations with prescribed cones [Bommes et al. 2013a, 2009; Bright et al. 2017; Chien et al. 2016; Ebke et al. 2016; Fu et al. 2015; Hefetz et al. 2019; Kälberer et al. 2007; Myles and Zorin 2012, 2013]. Interestingly, but not surprisingly, they do not follow the above two step principle—as no general method for the first step (valid initialization) is known for the arbitrary-topology arbitrary-cones setting. Instead, they are typically based on optimization subject to non-convex constraints and, despite long development and practical importance, no concise sufficient conditions for success are known. The key issue is that there is no available way to construct an initial solution, and one cannot guarantee that the solver will itself find a way into the feasible region.

Only for certain special cases there are known solutions in this regard, e.g., for specific genus or specific cones [Aigerman and Lipman 2015; Gu and Yau 2003], using more general non-piecewise-linear parametrization [Aigerman and Lipman 2016], or requiring additional input [Tong et al. 2006]. Particular challenges are caused by the fact that the given surface discretization may not even admit a (elementwise linear) solution, i.e., systematic remeshing capabilities are needed in any reliable approach.

Quadrangulation. The problem of surface quadrangulation with conforming elements and prescribed extraordinary vertices is closely related—state-of-the-art methods actually construct quadrangulations via seamless parametrization [Bommes et al. 2013b]. Jucovič and Trenkler [1973] investigate the question of existence of such quadrangulations. The result is purely combinatorial and does not yield a surface parametrization. On an abstract level, we adapt some of the general ideas in this work as foundation of our approach to modify non-seamless into seamless parametrizations through map padding.

In the context of quadrangulation, our strategy of transitioning from an initial non-seamless parametrization to a seamless one is, in a sense, similar to modifying a non-conforming quadrangulation into a conforming one. This has been tackled by simple subdivision or more involved T-mesh simplification techniques [Myles et al. 2014]—however, at the expense of not always preserving the prescribed extraordinary vertices. Our modification technique, by contrast, always preserves exactly the prescribed cones.

Cone selection. Regarding the (application specific) problem of choosing cone locations and curvatures, common approaches are based on considering surface curvature (e.g., via cross fields [Vaxman et al. 2016]), surface deformation [Marcias et al. 2013; Zhou et al. 2018], distortion reduction [Ben-Chen et al. 2008; Kharevych et al. 2006; Soliman et al. 2018; Vintescu et al. 2017], or on manual quadrangulation-driven design and editing [Campen and Kobbelt 2014; Ebke et al. 2016].

General holonomy prescription. Campen and Zorin [2017] address a related problem, showing that for any admissible *holonomy signature* one can construct (also via conformal maps) a *seamless similarity* map adequate for constructing T-splines. A holonomy signature, in addition to prescribed cone angles, includes turning angles around homology loops. In contrast, we use a stronger notion of seamlessness, not allowing scale jumps across cuts, while not controlling global turning angles around homology loops (cf. Section 7)—however, they are of the form $k\pi/2$ (for *some* k) by our construction.

3 SEAMLESS PARAMETRIZATION CONSTRUCTION

First, we define the notion of a seamless parametrization as well as a weaker notion of a *rotationally seamless* parametrization we need as an intermediate step.

Suppose a smooth surface M is cut to a set of topological disks M_i^c by a cut graph G, i.e., a collection of smooth curves (*branches*)

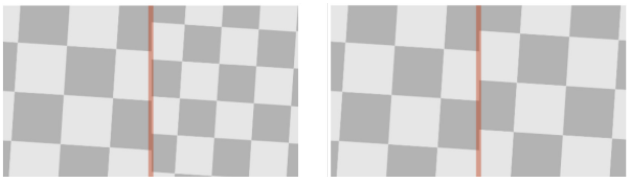


Fig. 3. Visualization of a parametrization on a surface near a cut branch (red). Left: rotationally seamless. Right: seamless.

γ_j embeddeded in M meeting only at their endpoints (*nodes*). We call the resulting cut surface M^c; the boundary of M^c consists of curves γ_j^c (boundary curves). There is a canonical map $\pi : M^c \to M$, which is identity in the interior of M^c and maps exactly two boundary curves γ_j^c to each branch γ_j on M. Pairs of boundary curves mapping to the same branch γ_j are called *mates*, and boundary points where curves γ_j^c meet are called *joints*. The image of any joint under π is a node. Pairs of non-joint points $p, q \in \partial M^c$ with $\pi(p) = \pi(q)$ are called *mated points*. For a boundary point p, let $t_p \in T_p M^c$ denote a unit vector that is tangent to the boundary ∂M^c at p.

Definition 3.1 (Rotationally Seamless Parametrization). A continuous, locally injective map $F : M^c \to \mathbb{R}^2$ is called *rotationally seamless* parametrization of M, if for any pair p, q of mated points, the images $dF_p(t_p)$ and $dF_q(t_q)$ of boundary tangents are related by a similarity transformation T_{pq}, i.e., $T_{pq} \circ dF_p(t_p) = dF_q(t_q)$, with a rotation angle that is a multiple of $\pi/2$ and constant per branch.

Such a rotationally seamless parametrization does, in general, have a (pointwise) *scale jump* across the cut (cf. Figure 3 left)—unless the similarity T_{pq} is actually just a rotation everywhere:

Definition 3.2 (Seamless Parametrization). A map $F : M^c \to \mathbb{R}^2$ is called a *seamless* parametrization of M, if it is rotationally seamless and for each pair of mated points p, q the transition T_{pq} is rigid, i.e., it is a rotation with a rotation angle that is a multiple of $\pi/2$.

Notice that seamlessness implies that the images $F(\gamma_j^c)$ and $F(\gamma_k^c)$ of mates γ_j^c and γ_k^c are congruent.

A seamless parametrization induces a metric on the surface M, which is flat except at the nodes, where it may be singular; it may have a *cone*. We say that a seamless parametrization has a cone with angle α at a node p, if the sum of parametric angles at all joints q in M^c with $\pi(q) = p$ is equal to α. This cone has curvature $\Theta = 2\pi - \alpha$.

By contrast, the notion of a rotationally seamless parametrization is weaker: Due to the scale jump, it does not induce a metric on M.

Overall Approach. We first construct a parametrization F that is rotationally seamless, using a specific type of (conformal) maps: maps with rectilinear boundary, i.e., with the image of the boundary of the cut surface consisting of straight segments meeting at right angles. Then this parametrization is modified near the boundary to make the scale jump vanish to make it into a seamless parametrization F^s. This is done using a process we call *map padding*. The key to our construction is cutting the surface into *two* (in special cases three or four) topological disks, using cut graphs with a particular structure. This is important for our method of converting rotationally seamless parametrizations into seamless parametrizations.

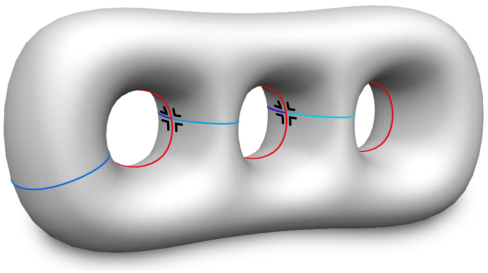
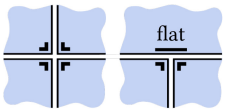


Fig. 4. Degree 4 cut graph on a surface of genus $g = 3$. This cut graph has 10 branches and 5 degree 4 nodes, thus 20 corners (marked black). The cut graph consists of loops (red) and connectors (shades of blue) (cf. Section 4.1).

3.1 Cutting to Disk(s)

We construct the required cut graph G in two steps, first cutting the surface M into a set of topological disks M_i'. Their disjoint union is denoted M'. Typically, we use two disks, with some exceptions for special genus 2 cases. M' contains all cones in its interior. The final cut surface M^c is obtained by adding branches passing through all cones to this cut graph, such that no M_i' is split into multiple components. This second step is explained in Section 3.3.

For the first step, we consider a particular type of cut graphs that only have nodes of degree 4 and 3. Pairs of cyclically sequential branches around nodes form sectors: four at degree 4 nodes, three at degree 3 nodes. At degree 4 nodes, all four sectors are marked as *corners* (cf. Figure 4). At degree 3 nodes, two sectors are marked as corners, the third one is referred to as *flat*. We refer to degree 3 nodes as *T-nodes*.



We denote the boundary curves of M' by γ_j'. Any pair of sequential boundary curves of M' corresponds to a corner or a flat joint. As we will require boundary curves to be straight and corners to have angles $\pi/2$ under a certain metric in the following, the number m_i of corners on the boundary of each connected component M_i' needs to match the total prescribed cone curvature in the interior of M_i' as per Gauss-Bonnet, i.e.,

$$m_i \frac{\pi}{2} + \sum_{(p_j, \hat{\Theta}_j) \in C_i'} \hat{\Theta}_j = 2\pi, \tag{1}$$

where $C_i' \subseteq C$ is the subset of cones prescribed within M_i'. Note that this is equivalent to $m_i = 4 + \sum_{(p_j, \hat{\Theta}_j) \in C_i'} (k_j - 4)$.

Definition 3.3 (Admissible Cut Graph). A cut graph with marked corners is *admissible*, if

- all branches are embedded smooth curves meeting transversally at nodes of degree 3 or 4, and not passing through cones;
- the cut graph partitions the surface into disk-topology components;

- the number of corners of each component satisfies Equation (1);
- if a boundary curve is involved in a flat sector, its mate is not.

3.2 Cone Metric with Rectilinear Boundary

Corners partition the boundary $\partial M'$ into *segments*. Note that a segment may contain flat joints, and, as a consequence, consist of several boundary curves γ'_j (*complex segment*). (All cut graphs we will be working with contain at most two T-nodes, thus two flat joints, i.e., almost all segments are simple segments.)

We now require a cone metric on M', which has a *rectilinear* boundary: under such a metric, segments are geodesically straight (i.e., zero geodesic boundary curvature along $\partial M'$ in the interior of segments), and sequential segments form right inner angles of $\pi/2$.

Proposition 3.1. *On a cut surface M', obtained from a smooth surface M by cutting it along an admissible cut graph G, there is a cone metric with rectilinear boundary and prescribed admissible cones $C = \{(p_i, \hat{\Theta}_i)\}$.*

A proof is given in Appendix C. In particular, a *conformal* cone metric with these properties exists; conformality, however, is not essential in the following. This cone metric on M' does not, in general, define a metric on M, as the lengths defined by the metric along the two sides of the cut graph's branches may disagree, cf. Figure 2, left.

Note that the metric angle on M around points on G is 2π everywhere: points in the interior of branches are surrounded by two sectors with angles $\pi + \pi$, node points by three sectors with angles $\pi + 2\frac{\pi}{2}$ or four sectors with angles $4\frac{\pi}{2}$. As we preserve these angles in the following, this implies that no spurious cones emerge.

3.3 Metric to Rotationally Seamless Parametrization

The cone metric with rectilinear boundary is flat away from the cones on M'. We now extend the cut graph G by a set of trees T_i, yielding the extended cut graph $G^T = G \cup T$, where T is the union of trees T_i. The tree T_i is rooted on $\partial M'_i$ at a single non-joint point, and its branches connect all cones prescribed within M'_i. Let M^c_i be the surface obtained by cutting M'_i along T_i. The segment of $\partial M'_i$ split by the root point of T_i is still considered one logical segment. Note that this cutting of M' to M^c by T introduces additional boundary curves; we do *not* refer to these as segments, and they are, in general, not straight under the cone metric. The boundary curves of M^c are denoted γ'_j if they map to branches of G, or γ^T_j if they map to branches of T. The above constructed cone metric is flat in the interior of M^c_i (as the cones lie on ∂M^c), and defines (via integration) a map $F_i : M^c_i \to \mathbb{R}^2$. It is unique up to a rigid transformation. We choose this transformation so that all segment images are axis-aligned in $\mathbb{R}^2$. This is possible because they (due to rectilinearity) are all straight and meet at right angles. Note that images of boundary curves γ^T_j are, in contrast to segments (consisting of boundary curves γ'_j), neither axis-aligned nor straight in general, cf. Figure 18 in Appendix A. Together, these maps F_i define a global parametrization F of M.

Proposition 3.2. *The map F is a rotationally seamless parametrization of M (but not, in general, seamless—except on T).*

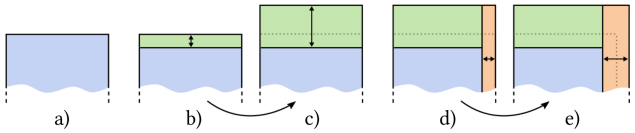


Fig. 5. (a) Generic local view of the boundary of map $F(M^c)$, with straight segments and right-angle corners. (b) A rectangular strip along a segment is marked. (c) The strip is stretched outwards, effectively increasing the length of the two adjacent segments left and right of the central segment. (d) This padding operation can be applied in sequence to further segments.

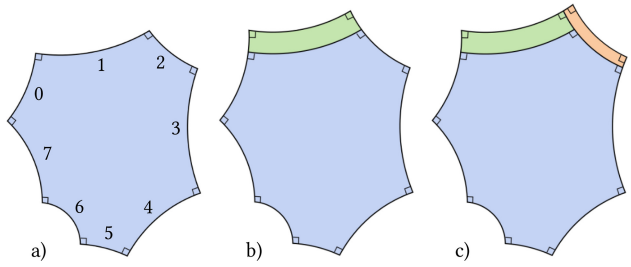


Fig. 6. (a) Global visualization (without cuts to cones) of the rectilinear map, where straight segments appear as curved arcs (as explained in Section 3.3). (b) Padding (analogous to Figure 5) of segment 1, increasing the lengths of segments 0 and 2. (c) Padding of segment 2, increasing the lengths of segments 1 and 3. This can be continued to adjust all segments' lengths.

Proof. Due to all segment images being axis-aligned, the angle between the images of any two mated boundary curves γ'_j, γ'_k is some multiple of $\pi/2$, constant per branch. The images of any two mated boundary curves γ^T_j, γ^T_k are congruent (in particular, similar) as the metric is flat on T by construction. The rotation between them is a multiple of $\pi/2$, because the prescribed angles at cones are multiples of $\pi/2$ (cf., e.g., Springborn et al. [2008]). Hence, F is seamless on T but, in general, only rotationally seamless on G. □

Visualization. For purposes of illustration, we would like to visualize the image $F(M^c)$. Due to global overlaps implied by negative curvature cones, this is not an easy task. However, locally, near the cut graph G, $F(M^c)$ always looks like in Figure 5(a)—because the boundary consists exclusively of straight segments meeting at right-angle corners. (The only exception being the one boundary curve per M_i where the tree T_i is rooted.) We use this type of illustration when a local view is sufficient. An alternative is to flatten the surface globally, without cutting to the cones, instead (for visualization purposes) pushing the curvature of the cones evenly onto the boundary $\partial M'$. This leads to a flattening of M'_i as shown in Figure 6(a), where straight boundary segments appear as curved arcs (and cones are not visible). This makes it possible to visualize the complete rectilinear boundary without cuts or overlaps.

3.4 Seamless Parametrization by Padding

The rotationally seamless map F differs from a seamless map in two respects: the images of mated segments may have different lengths, which implies a scale jump; but even if they are of equal length, this only implies that the scale is equal on average, rather than pointwise along the corresponding branch.

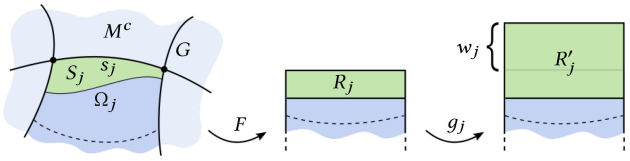


Fig. 7. Illustration of strip definition and stretch map applied to perform padding of a segment s_j by padding width w_j, cf. Section 3.4.

We thus modify F by composing it with two types of local segment-wise maps:

- a *stretch* map g_j, which effects a change of the lengths of segment images,
- a *shift* map r_j, which subsequently equidistributes scale along a segment.

We iteratively apply these operations segment by segment. Note that these are applied only to *segments*; across the additional cuts T the map F is already seamless, cf. Proposition 3.2.

Stretch. For a boundary segment s_j, we consider a thin strip S_j on M^c, which runs along the entire segment and maps to a rectangular region R_j via F. This is illustrated in Figures 5(b) and 7.

More formally, the strips are defined as follows. The restriction of F to the segment s_j (which maps s_j to a straight segment in the plane) is bijective, and so is the restriction F_j to a sufficiently small neighborhood $\Omega_j \subset M^c$ of s_j. We choose the rectangle R_j within $F(\Omega_j)$ such that it includes $F(s_j)$ but no cone points and no joints except the ones on s_j. The thin strip S_j on M^c is then defined as $S_j = F_j^{-1}(R_j)$, as shown in Figure 7.

Outside of S_j, we preserve the map, but within S_j, we modify it by a one-dimensional scale map g_j such that S_j is mapped onto a larger rectangle $R'_j \supseteq R_j$ whose width (orthogonal to the segment s_j) is increased by a *padding width* w_j. This is illustrated in Figures 5(c) and 7. Effectively, the domain is locally *padded* by an additional rectangular region $R'_j \backslash R_j$ of width w_j along the image of s_j, cf. Figure 6.

The situation is slightly different at the one segment s_j per component where T is rooted (cf. Section 3.3): it is separated into two parts by T (cf. Figure 20, left). Both parts can, however, be handled separately using the same technique, as detailed in Appendix D.

We define the padded map F^p iteratively, iterating over the (arbitrarily ordered) strips S_j, $j = 1, \ldots, n$, of each connected component of M^c. $F^{p,0}$ coincides with F, and $F^{p,m+1}$ differs from $F^{p,m}$ only on R_{m+1}, where it is defined as $g_{m+1} \circ F^{p,m}|_{R_{m+1}}$, where g_{m+1} is the above scaling transformation (detailed in Appendix D). $F^p = F^{p,n}$.

Shift. In the padded map F^p, we consider for each segment a strip S_j^p (now defined based on F^p) and modify the map within this strip by a map r_j with the following properties:

- its restriction to a simple segment s_j reparametrizes s_j to constant speed; for complex segments: constant speed per contained boundary curve, cf. Appendix D,
- the map is equal to identity on the rest of the strip's boundary,
- it is continuous and bijective.

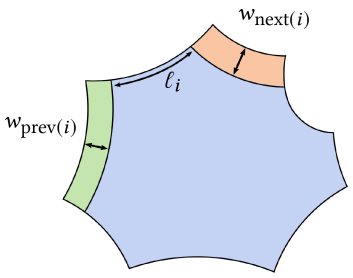


Fig. 8. The length of segment i is affected by the padding of the two adjacent segments: the original length ℓ_i changes to $\ell_i + w_{\text{prev}(i)} + w_{\text{next}(i)}$.

We define the shifted map F^s iteratively, again iterating over the strips S_j^p of each connected component of M^c. $F^{s,0}$ coincides with F^p, and $F^{s,m+1}$ differs from $F^{s,m}$ only on $R^p_{m+1} = F^{s,m}(S^p_{m+1})$, where it is defined as $r_{m+1} \circ F^{s,m}|_{R^p_{m+1}}$, where r_{m+1} is the above shift map (detailed in Appendix D). Finally, $F^s = F^{s,n}$.

Proposition 3.3. *F^s is a rotationally seamless parametrization of M with the same cones as F, and not only the angle but also the scale jump is constant per branch of the cut.*

A proof is given in Appendix D. The choice of padding widths determines the lengths of segments under F^p and, as they are preserved by the shift maps, under F^s. In the following Section 3.5, we detail how *equalizing* padding widths can be found:

Definition 3.4 (Equalizing Padding Widths). A set of padding widths leading to all pairs of mates being of equal length under F^s is called *equalizing padding widths*.

Proposition 3.4. *If F^s is constructed using equalizing padding widths, then it is a seamless parametrization of M.*

Proof. The constant scale jump per branch (Proposition 3.3), together with the equal lengths of mated boundary curve images implies a scale jump of zero per branch, i.e., rigid transitions. □

3.5 Length Equalization

When a boundary segment is padded, the lengths of the two adjacent segments' images change. To make this precise, let ℓ_i be the length of segment s_i before any padding is performed, and w_i be the amount of padding applied to s_i. The length ℓ'_i of s_i after each segment was padded according to values $\boldsymbol{w} = (w_0, w_1, w_2, \ldots)$ is

$$\ell'_i = w_{\text{prev}(i)} + \ell_i + w_{\text{next}(i)}, \tag{2}$$

where prev(i) and next(i) are the two segments adjacent to s_i, preceding and following it in cyclic order along $\partial M'$, cf. Figure 8.

Our goal is to find an equalizing assignment of padding width variables $\boldsymbol{w}$ such that the lengths $\boldsymbol{\ell'} = \{\ell'_0, \ell'_1, \ell'_2, \ldots\}$ after padding are equal for each pair (s_i, s_j) of mated segments, i.e., $\ell'_i = \ell'_j$. This leads to *length equalization* equations

$$w_{\text{prev}(i)} + w_{\text{next}(i)} - w_{\text{prev}(j)} - w_{\text{next}(j)} = \ell_j - \ell_i. \tag{3}$$

However, only if all cut graph nodes are of degree 4, all segments are simple, and mated in pairs as a consequence. In the presence of T-nodes (which imply complex segments) the situation is a little different: a complex segment s_i consists of multiple boundary

curves; their mate curves, however, form simple segments due to the last property of Definition 3.3. Hence, generally, a (simple or complex) segment s_i is mated with a sequence $J_i = (s_j, s_k, \ldots)$ of one or more simple segments. Length equalization equations then take this more general form:

$$\ell'_i = \sum_{j \in J_i} \ell'_j, \tag{4}$$

which expands to

$$w_{\text{prev}(i)} + w_{\text{next}(i)} - \sum_{j \in J_i} \left(w_{\text{prev}(j)} + w_{\text{next}(j)} \right) = \sum_{j \in J_i} \ell_j - \ell_i. \tag{5}$$

These equations form a globally interdependent equation system:

$$A\boldsymbol{w} = \boldsymbol{b}, \quad w_i \geq 0 \, \forall i. \tag{6}$$

Note the non-negativity condition; it ensures that the padding operation actually stretches ($w > 0$) rather than squeezes ($w < 0$) the strips along segments. Allowing squeezing would require an upper-bound constraint.

This system needs to be solved to achieve length equalization of mated segments, enabling seamlessness. Unfortunately, it is not generally feasible—it may have no non-negative solution or even no solution at all. Notice that the system matrix structure is entirely determined by the cut graph's combinatorics, leading to the following definition.

Definition 3.5 (Equalizable Cut Graph). An admissible cut graph for which equalization system Equation (6) is feasible for arbitrary $\boldsymbol{b}$, is called *equalizable*.

Our key result, accompanying the proposed map padding technique, is a proof showing that there is an equalizable cut graph for any genus and any admissible set of cones, as well as an efficient algorithm to construct such cut graphs (cf. Section 4 and Appendix B).

4 EQUALIZABLE CUT GRAPHS

We prove the following proposition:

Proposition 4.1. *For any genus g and any admissible prescription of cones C, there is an equalizable cut graph; i.e., we can always obtain equalizing padding widths that enable a seamless parametrization.*

The foundation of our construction of equalizible cut graphs is a graph we call a *hole chain*. Variations thereof, depending on the surface's genus and the cone configuration, then yield equalizable cutgraphs. Cut graphs covering all cases (arbitrary genus, arbitrary admissible cones) are defined in this section. Their equalizability is shown in Appendix B.

Together with Proposition 3.1 (rectilinear cone metric), Proposition 3.2 (rotationally seamless map), and Proposition 3.4 (seamless modification given equalizing padding widths), this Proposition (equalizable cutgraphs) **concludes the constructive proof of the main theorem 1.1.**

4.1 Hole Chain

Given a closed surface M of genus $g > 0$, we cut it along g non-intersecting non-homotopic non-separating smooth loops α_i. This yields a topological sphere M° with $2g$ holes. Note that each loop

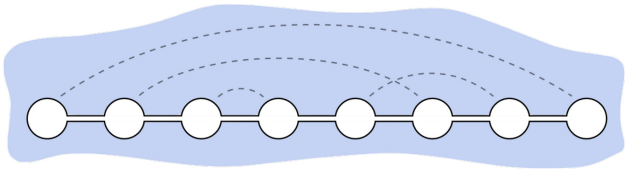


Fig. 9. Schematic depiction of a chain of holes for a genus $g = 4$ surface: circles are holes (obtained by cutting the surface along g loops), straight line segments are the sides of cut paths (connectors) between these holes. Together, the hole chain cuts the surface to a topological disk (blue), i.e., a sphere with one hole (white, bounded by the black curve). An example of a hole partner correspondence is indicated by dashed arcs; depending on the chosen ordering of holes in the chain, these partner arcs will look different.

corresponds to two holes, which are called *partners*. Let the holes be numbered from 0 to $2g - 1$, and denoted h_i, in such a way that h_0 and h_{2g-1} (called *terminals*) are partners.

Let $\pi : M^\circ \to M$ be the canonical map from M° to M, taking h_i and its partner h_j to their corresponding loop α: $\pi(h_i) = \pi(h_j) = \alpha$.

On each hole h_i, pick two distinct points q_i and q'_i, such that they are identified across partners on M; i.e., for partners h_i, h_j, we have $\pi(q_i) = \pi(q_j)$ and $\pi(q'_i) = \pi(q'_j)$. For each $0 \leq i < 2g - 1$, we further cut M° along a smooth non-intersecting path between holes h_i to h_{i+1}, starting transversally at point q'_i and ending transversally at point q_{i+1}. These paths are called *connectors*. Note that after each such cut the surface remains a topological sphere with holes (decreasing in number), thus, it remains path-connected; therefore, these connectors always exist. Loops and connectors together form a cut graph we call *hole chain*, as depicted abstractly in Figure 9 and concretely in Figure 4, which yields the surface M', a sphere with one hole, i.e., a disk. Loops and connectors are assumed not to intersect any prescribed cone.

Proposition 4.2. *The hole-chain cut graph for any genus $g > 0$ is admissible.*

Proof. As the connectors' endpoints q_i, q_j are identified in pairs on M across partners h_i, h_j, each resulting cut graph node (at point $\pi(q_i) = \pi(q_j)$ on M) is of degree 4 (cf. Figure 4). All branches are smooth curves meeting transversally at their endpoints and not crossing cones by construction. The surface is cut to a single component with disk topology. As the set of cones C is admissible, we have $\sum_j \Theta_j = 2\pi(2 - 2g)$; the hole chain cut graph has $2g - 1$ nodes with 4 corners each, i.e., $8g - 4$ corners. Thus, Equation (1) is satisfied. □

Odd-Couple Condition. We impose one condition (besides terminals being partners) on the way the numbering of holes is chosen: there needs to be at least one odd couple, i.e., two partner holes that have an odd number of holes between them in the chain, i.e., there is an i and an integer k such that h_i and h_{i+2k} are partners. This will be expected in the proof of equalizability. Note that this is impossible if there are just four or less holes, thus instead special case variations of the hole chain are used for genus 1 (two holes) and genus 2 (four holes) cases, as detailed in Section 4.3.

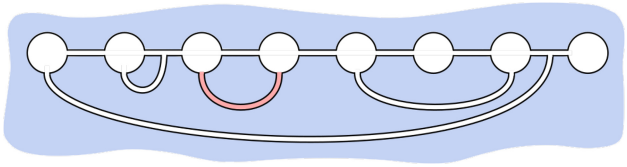


Fig. 10. Examples of extra paths (bottom) that could be added to the hole chain cut graph. The red path is not an admissible extra path, because it splits the surface into two components with $m_0 = 4$ and $m_1 = 8g - m_0 = 28$ corners (cf. Section 4.2).

Definition 4.1 (Fourfold Cones). A set of cones $C = \{(c_i, k_i)\}$ with k_i divisible by 4 for each i is called *fourfold*.

While the above hole chain cut graph is not equalizable in general, it permits equalization for *specific* righthand sides $\boldsymbol{b}$:

Proposition 4.3. *For any genus, a fourfold cone prescription implies a righthand side $\boldsymbol{b}$ for which Equation (6) is feasible.*

A proof is given in Appendix B. For the general, non-fourfold case, variations of the hole chain are used. We will describe the necessary changes next.

4.2 General Case (Genus 3+)

In case the cone prescription is not fourfold, i.e., there is at least one $k_i \mod 4 \neq 0$, we extend the hole chain cut graph by one extra path (cf. Figure 10)—which makes it equalizable.

Definition 4.2 (Valid Extra Path). A simple path is called a valid extra path for a hole chain cut graph if

- it does not pass through a cone,
- only its endpoints are contained in the hole chain cut graph,
- at least one endpoint is on a hole of the hole chain,
- no endpoint is coincident with a node of the hole chain.

Notice that this extra path forms two additional nodes, both of degree 3, i.e., T-nodes (or, if coincident, one of degree 4) at its endpoints. At each, we mark as corners the two sectors directly adjacent to the extra path. Hence, the total number of corners increases from $8g - 4$ to $8g$. At the same time, the extended hole chain cut graph cuts the surface into two components, M_0' and M_1', each with disk-topology, with numbers of corners m_0, m_1. In particular, we have $m_0 + m_1 = 8g$.

For the extended hole chain to remain admissible, we need to ensure that Equation (1) is satisfied, i.e., the number m_i of corners per component M_i' needs to match the total curvature of cones C_i' prescribed within the component. Note that this is satisfied for M_0' if and only if it is satisfied for M_1'. Also, if the endpoints lie on two mated segments, we choose them as mated points, to create a degree 4 node rather than two opposite T-nodes. We furthermore require that the numbers m_0, m_1 of corners are not divisible by 4. This will be expected in the proof of equalizability.

Definition 4.3 (Admissible Extra Path). A valid extra path that yields corner numbers m_0, m_1 not divisible by 4 and satisfying Equation (1) is called admissible.

Proposition 4.4. *An admissible extra path exists for any non-fourfold cone prescription and any genus $g \geq 3$.*

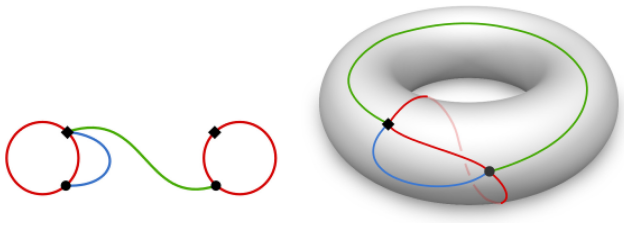


Fig. 11. Cut Graph pattern for genus 1 surfaces, shown abstractly (left) and on an example surface (right). The surface is partitioned into a 2-corner region (enclosed by blue and red paths) and a 6-corner region.

Proof. Pick one prescribed cone c_i with $k_i \mod 4 \neq 0$. Let β be a simple path from a non-corner point q on the hole h_0 to c_i, not containing any other cone. Let γ be a path that starts at q, runs (arbitrarily close) along one side of β, then around c_i, then back along the other side of β, and ultimately (arbitrarily close) along the cut G until it has passed $k_i - 2$ corners. It then connects to a point q' on the segment it reached. If γ is chosen sufficiently close to β and G, then it is an example of a valid extra path: the region that contains c_i contains no other cone and it has k_i corners (the $k_i - 2$ corners passed along G plus the two corners formed by the extra path itself with G at q and q'). Also, γ is connected to a hole, namely h_0. □

The equalizability of the hole chain cut graph with an odd-couple extended by an admissible extra path is proven in Appendix B.

4.3 Special Cases (Genus 0, 1, 2)

Genus 0 Case. In the case of a topological sphere, our method formally is applicable, but does not actually contribute anything: the cut graph is empty; there are no cuts across which the cone metric could be non-seamless, thus no padding is required. The existence of conformal metrics with prescribed cones on the topological sphere M is well-known [Troyanov 1991].

Genus 1 Case. For genus 1 surfaces, we (similar to the general case) add one extra path, but deviate slightly from the general hole chain pattern in terms of identification of connector endpoints. The cut graph pattern is depicted in Figure 11. Notice that the surface is split into two components, with 2 and 6 corners. It is easy to see that for any admissible prescription of cones on a genus 1 surface, one either has no cones at all (in this case the basic hole chain cut graph is sufficient, cf. Proposition 4.3) or one has, among the prescribed cones, one or more cones whose curvature sums up to π (due to the Gauss-Bonnet theorem, there are cones of positive and negative curvature, and the case of a single positive cone of curvature $\pi/2$ is the one non-admissible case, cf. Section 1). The surface bi-partition by this cut graph pattern is thus compatible with any non-empty cone prescription, i.e., the paths can be chosen in an admissible way on M.

The equalization equation system of this pattern is easily checked explicitly for non-negative feasibility (cf. Appendix B.2).

Genus 2 Case. For genus 2 surfaces, we also need to deviate from the general case. In contrast to the genus 1 case, where we could assume that a subset of prescribed cones always have curvatures

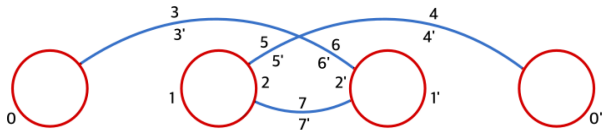


Fig. 12. One of the cut graph patterns for genus 2 surfaces. Segments i and i' are mates, i.e., correspond to a common cut graph branch. The surface is partitioned into a 5-corner region (center) and a 11-corner region (surround).

summing up to one specific value, five cases need to be distinguished: there is a subset of prescribed cones with curvatures summing to π, $\pi/2$, $-\pi/2$, $-\pi$, or $-3\pi/2$ (compatible with regions with 2, 3, 5, 6, and 7 corners, respectively). This list is exhaustive, because the total sum of cone curvatures is -4π on a genus 2 surface, and not all cones have curvatures that are multiples of 2π (as this case was handled already in Proposition 4.3); as a consequence, at least one of these five values has to appear as a subsum.

Depending on which curvature sum subset is available in a given set of prescribed cones, the cut graph pattern needs to be chosen compatibly. For the case that a cone subset with curvature sum $-\pi/2$ is available (as in most practical scenarios), the pattern depicted in Figure 12 can be used; for the remaining patterns refer to Appendix B.3. Notice that this pattern is a variation of the basic hole chain: two connectors are required to cross. This partitions the surface into a 5-corner region (compatible with a curvature $-\pi/2$ subset) and a 11-corner region (compatible with the remaining cones).

The equation systems corresponding to these patterns are easily checked explicitly for non-negative feasibility.

5 IMPLEMENTATION

We now describe how our algorithm can be implemented for discrete surfaces. In this section, M denotes a closed triangle mesh of arbitrary genus g. Cones are prescribed at vertices of M; such vertices are called *cone vertices*. We first focus on the case $g \geq 3$, then on the minor deviations required for cases $g = 1$ and $g = 2$.

General Overview. The main steps of the algorithm are:

(1) Construct g non-contractible loops and cut M along these loops (Section 5.1).
(2) Connect all holes using shortest paths, selecting the connection pattern based on the given genus and cones. Where necessary, add one extra path; then cut the mesh (Section 5.1).
(3) Set target angles at cone and corner vertices. Compute the corresponding discrete conformal metric with prescribed curvature (Section 5.2).
(4) Number all cut segments and set up the system matrix A and righthand side $\boldsymbol{b}$ accordingly (Section 5.3).
(5) Compute a solution to the linear system $A\boldsymbol{w} = \boldsymbol{b}$; add a constant shift to yield a solution $\boldsymbol{w} \geq 0$ (Section 5.3).
(6) Extend the cut graph to include all cones; lay out the mesh in the plane according to the metric (Section 5.4).
(7) Perform padding according to padding widths $\boldsymbol{w}$ (Section 5.5).

The output of the algorithm is a seamless parametrization with the prescribed cones. While the parametric distortion initially is high near the cuts, this parametrization provides the *feasible starting point* required by techniques for injectivity-preserving parametrization distortion optimization (cf. Section 5.6) as well as quadrangulation methods based on quantization of seamless parametrizations.

5.1 Cut Graph

On M, one option to obtain g non-intersecting non-contractible loops is via handle/tunnel loop algorithms [Dey et al. 2013], modified to avoid cone vertices. A simpler robust approach is to iteratively cut the mesh g times, each time along an arbitrary non-contractible loop not containing a cone or a boundary vertex, obtained using the tree-cotree algorithm [Erickson and Whittlesey 2005]. To ultimately yield a cut graph that is not unnecessarily convoluted, it is advisable to pick a short loop each time.

We construct the connectors between the $2g$ holes as shortest paths, not containing cone, boundary, or other paths' vertices, using Dijkstra's algorithm. A natural ordering of the holes can be determined using a Hamiltonian path algorithm; this order needs to be adjusted slightly before connector construction, to ensure the paired-terminals and odd-couple conditions are satisfied (cf. Section 4.1).

Mesh Refinement. We work with discrete paths/loops, following the edges of M. For the algorithm to be robust regardless of the mesh structure, after each construction of a path, we split each mesh edge that is not on a path if its two vertices both are either on the boundary, on a path, or on a cone. This ensures that the mesh, cut by the paths, remains path-connected with respect to the discrete edge paths, avoiding boundary, path, and cone vertices.

Extra Cut. The extra cut path (needed for the general genus $g \geq 3$ case) is constructed as a shortest path as well. However, we need to employ a cone-aware variant of Dijkstra's algorithm to ensure cone/corner compatibility, cf. Equation (1).

To this end, to each directed dual edge e of the mesh M', we assign a value ρ_e (with $\rho_e = -\rho_{\bar{e}}$ for oppositely directed dual edges e, $\bar{e}$) such that the sum of these values clockwise around a single cone vertex c_i is $\hat{\Theta}_i$, and around non-cone vertices zero. Such an assignment can, for instance, be obtained using a spanning tree of the cones (Figure 13), rooted at the boundary $\partial M'$: initialize ρ at all leaves of the tree and propagate the values towards the root, summing values where branches meet.

Now for an arbitrary simple closed clockwise dual edge path γ, we have the following important property: $\sum_{e\in\gamma} \rho_e = \sum_{p_i\in\Gamma} \hat{\Theta}_i$, where Γ is the set of all vertices enclosed by γ [Crane et al. 2010]. We call this sum $\sum_{e\in\gamma} \rho_e$, with a slight abuse of terminology also for non-closed paths, *(partial) holonomy*. For a closed path, the sum $\sum_{e\in\gamma} \rho_e$ along a closed path tells us what total cone curvature is contained in the region enclosed by the path.

We then employ Dijkstra's algorithm, starting from a hole segment on $\partial M'$, and keep track of the partial holonomy values along the way. Whenever the front propagation in Dijkstra's algorithm reaches $\partial M'$ again, we tentatively close the loop by walking back to the starting point clockwise along $\partial M'$, counting passed corners

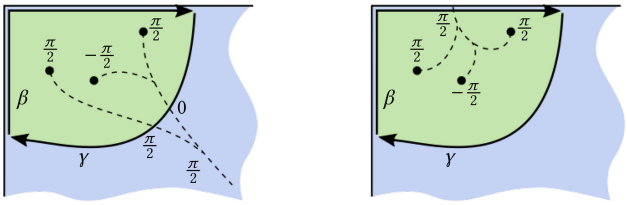


Fig. 13. Example of the holonomy-aware extra path computation. Left: a tree of cones with computed ρ-values is shown as black dashed lines. Path γ from boundary to boundary, crossing two tree branches, has a holonomy value $\sum_\gamma \rho = \pi/2$. This path is closed along the boundary by β (with $\sum_\beta \rho = 0$), forming $m = 3$ corners. As $\sum_{\gamma+\beta} \rho = \pi/2$ and $m = 3$ conforms with Gauss-Bonnet (1), the path γ is admissible. Right: to illustrate that the tree of cones can be chosen arbitrarily, here the same situation is depicted with a different tree. We have $\sum_\gamma \rho = 0$ and $\sum_\beta \rho = \pi/2$, thus again $\sum_{\gamma+\beta} \rho = \pi/2$.

on the way, and checking whether the total holonomy matches the number of corners, cf. Section 3.1, Equation (1). If it matches, then the path is accepted and added as extra cut. An example is shown in Figure 13.

A modification is needed to the standard algorithm, though: Dijk-stra's algorithm keeps track of, for each vertex, the shortest path back to the starting point—regardless of partial holonomy $\sum \rho$. So while there are shortest paths of different partial holonomy back to the starting point, Dijkstra's algorithm discards all but the shortest one. We, instead, keep track of the shortest path *per vertex per holonomy value*. Otherwise paths that could end up having a suitable holonomy in the end, may be discarded early. We therefore perform Dijkstra's algorithm not on M', but on a branched covering of M' [Kälberer et al. 2007], with sheets glued according to ρ. In practice, this means that each triangle stores separate distance information per partial holonomy value, indexed by $\sum \rho$ of incoming fronts.

We also need to ensure that γ is simple on M'. While the holonomy-aware version of Dijkstra's algorithm yields a simple path on the covering, its projection to M' may be self-intersecting. Before advancing the front to the next vertex, we always check whether this vertex is already contained in the predecessor path to prevent such self-intersections. While with this latter modification it is no longer guaranteed that a path is always found, one can always fall back to an explicit path construction following the existence proof in Section 4.2; we have never encountered a case where this was necessary.

Special Cases. The connectors of the special cut graph patterns employed for genus 1 and genus 2 surfaces are realized using shortest paths as well. These are constructed incrementally between endpoints chosen on the holes, and cross points chosen on other connectors where necessary. For those paths that split the surface into disjoint components, again the above cone-aware shortest path algorithm is employed to ensure cone/corner compatibility.

5.2 Conformal Map

After cutting M using the cut graph G to obtain M', for each component of M' (typically one or two, except for some genus 2 cone configurations), we need to obtain a cone metric with rectilinear boundary. This is the one part of the implementation, where achieving robustness in the discrete case is a challenge, even though in the smooth case (cf. Section 3.2) things are rather straightforward.

In contrast to the continuous case, questions of existence of (some notion of) *discrete* conformal metrics with prescribed cones and boundary curvature are not fully settled. For cases without boundary, recent results have brought theoretical insights [Gu et al. 2018a, 2018b; Luo 2004; Springborn 2017] and provide an algorithmic foundation but the boundary case requires further work on the theory side.

We use a modification of the conformal mapping algorithm for meshes without boundary described in Campen and Zorin [2017]; it combines the elegant variational formulation of Springborn et al. [2008] with on-the-fly mesh modifications (edge flips, following [Luo 2004]). For the genus 0 case, this algorithm can be used directly (cf. Section 4.3) to produce a seamless parametrization. With a minor extension, we additionally prescribe geodesic boundary curvature, using the holonomy angle constraints offered by this method, applied for each boundary vertex's triangle fan. The angle values are set to $\pi/2$ at corners and π at all other boundary vertices. While this algorithm is observed to behave well in practice, as mentioned earlier, further work is necessary to determine whether formal guarantees (regarding general existence and convergence) can be established.

In this context let us remark that our overall seamless parametrization construction does not in any way rely on the cone metric actually being conformal—this was merely a convenient natural choice that allows to easily ensure existence of a locally injective map with a given boundary curvature in the proof. If different approaches to conformal mapping, to other discrete notions of conformal mapping, or entirely different non-conformal methods for parametrization with piecewise straight boundaries are designed, then these can be used alternatively.

A recently proposed conformal mapping method [Sawhney and Crane 2017] supporting cone and boundary curvature prescription is particularly efficient—but, for instance, does not include remeshing capabilities inevitably required for full robustness. One could construct a hybrid solution with the more efficient algorithm tried first, and the robust but slower one serving as fallback.

Note that if this is important in a use case, then the edge flips that are performed by the conformal metric computation algorithm can ultimately be realized by means of edge splits, as described in Fisher et al. [2007]. In this way the output mesh is a locally refined version of the input mesh (rather than a mesh with arbitrarily different combinatorial structure) and its embedding is preserved.

5.3 Equalization

We (arbitrarily) number the segments of $\partial M'$ and set up the system matrix A Equation (6) accordingly, with one Equation (3) for each pair of mates (or Equation (5) where T-nodes are involved, cf. Section 4.2). The right-hand side $\boldsymbol{b}$ is determined by measuring the lengths of the segments under the metric computed in Section 5.2.

Then, we solve the linear system $A\boldsymbol{w} = \boldsymbol{b}$. As it is underdetermined, we compute the least-norm solution $\boldsymbol{w}^*$ via $A^T A\boldsymbol{w} = A^T\boldsymbol{b}$.

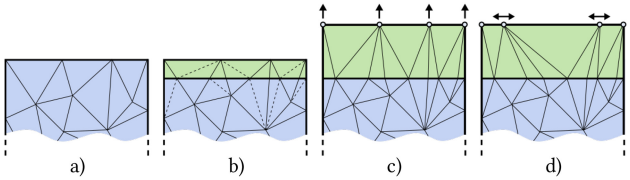


Fig. 14. (a) Mesh near a segment (top) to be padded. (b) The strip to be stretched (green) is formed by inserting a straight line into the triangulation (by splitting edges at the intersections), so close to the segment that no vertex is contained. (c) The strip is stretched outwards by displacing the vertices that lie on the segment by the desired padding width. (d) The vertices on the segment are translated laterally according to ϕ for pointwise seamlessness.

If the cut graph contains an extra path (cf. Section 4.2), then we additionally fix the two padding width variables associated with the two extra path segments to zero (by eliminating them from A); Proposition B.1 asserts that the system remains feasible.

The resulting solution $\boldsymbol{w}^*$ does not generally satisfy the important non-negativity constraint of Equation (6). However, we can now add a sufficiently large constant (as described in the proof of Proposition B.1) to all padding widths w_i^* (except those associated with an extra path, already fixed to zero). This ensures that the equalizing padding widths are non-negative.

5.4 Flattening

To obtain the map F from the conformal cone metric, we first need to extend the cut graph G to G^T (cf. Section 3.3). In each component of M', we pick a non-corner point p on a segment, and compute the shortest paths from p to all cones in the component. The union of these paths forms the tree T, extending G to G^T. Note that the piecewise-linear form of padding we use in the discrete setting (cf. Section 5.5) does not require the tree to meet the segment at right angles at root point p.

The conformal metric computed in Section 5.2 is flat on all of M^c, so its components can be laid out in the plane [Springborn et al. 2008], isometrically with respect to the metric, to obtain $F(M^c)$.

5.5 Padding

The padding operation described in Section 3.4 can be performed using piecewise-linear maps, as illustrated in Figure 14. We start with inserting a straight (in flat/parametrization metric) line into the mesh M along a segment, requiring that no vertex is contained in the strip this line delineates. This line cuts the edges and creates new vertices; resulting polygons are triangulated. Then, stretch and lateral shift can be performed by moving those vertices that are on the segment. Note that, as no vertices lie in the strip, the mesh within the rectangular strip is a simple triangle strip and laterally translating the segment vertices does not cause triangle inversions as long as their order is preserved—which is the case with the reparametrization ϕ (cf. Appendix D).

We note that the resulting seamless parametrization is not of immediate practical use: The scale distortion involved in the conformal map together with the additional padding-induced stretching often leads to high parametric distortion. This map, however, provides a valid (locally injective and seamless) starting point required by robust optimization methods that can convert it to a low-distortion parametrization (to the extend permitted by the cone prescription). We emphasize that for non-convex problems, the ability to obtain a feasible starting point is critical: first, this is, in general, the only way to guarantee that a solution is found; second, this allows one to use robust optimization techniques that always stay in the feasible region during optimization.

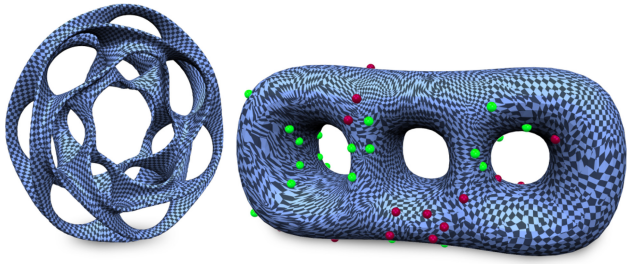


Fig. 15. Left: example map generated on a topologically complex surface. Right: Example map generated with geometrically non-meaningful cone prescription (here: 50 randomly distributed cones of curvatures π and $-\pi$) to illustrate the method's robustness.

5.6 Distortion Optimization

For the optimization of the seamless padded map, in our implementation, we use the symmetric Dirichlet energy together with efficient quadratic proxies as described in Rabinovich et al. [2017]. This method preserves local injectivity during optimization by design; we additionally include linear seamlessness constraints to preserve seamlessness of the map:

$$\vec{e_i} = R^{k_{ij}\frac{\pi}{2}}\vec{e_j} \quad \text{for each pair } (i,j) \text{ of identified mesh edges,}$$

where $\vec{e_i}$ is the edge vector of edge i in the parametric domain, and $R^{k_{ij}\frac{\pi}{2}}$ is a rotation by $k_{ij}\frac{\pi}{2}$, where the constant integer k_{ij} is determined by the edges' relative initial orientation in the domain.

We use the common symmetric Dirichlet objective in the optimization. In the context of specific applications, other application-dependent objectives, not focussing on distortion alone, may be relevant. For instance, for seamless global texturing, the seamless parametrization would have to be quantized (resulting in particular in an *integer grid map* with discrete translations in the transitions across cuts) [Ray et al. 2010]. The same is true for quad mesh generation or for constructing domains for spline spaces. Techniques to perform such quantization rely on using seamless parametrizations as a starting point [Campen et al. 2015; Lyon et al. 2019], of the type our method provides. It can also be useful to include directional terms in the objective, to support alignment to principal curvature or other directions. Finally, potential inaccuracies in the results' seamlessness due to numerical precision limits could be eliminated [Mandad and Campen 2019].

6 EXAMPLES

We demonstrate our implementation of the algorithm described in Section 5.1–5.6 on a number of examples.

Figure 16 shows a visualization of the seamless parametrizations constructed on models from the dataset provided by Myles et al. [2014]. We employed the cone position and angle prescriptions included in this dataset. Figure 15 demonstrates the

Fig. 16. Visualization of a variety of locally injective seamless parametrizations obtained using our method. Note that the cut is visible in the checkerboard texture, because the seamless parametrization is not a quantized seamless parametrization.

Fig. 17. A locally injective seamless map generated on an 80-torus.

algorithm handling topologically complex surfaces as well as randomly prescribed singularities. In all cases locally injective seamless global parametrizations were obtained. Note that seamless does not mean that cuts are not visible in these checkerboard visualizations; for this the maps would additionally need to be quantized [Campen et al. 2015]—a process for which our method provides a suitable initialization.

To explore the numeric limits of our implementation, we applied it to N-tori, for increasing N. For an 80-torus, as depicted in Figure 17, the implementation succeeds; for a 100-torus, we are still able to obtain an initial seamless map—however, with a level of distortion that state-of-the-art local injectivity preserving optimization methods prove to have trouble with, due to numerical precision issues. For even larger N, the computation of the constrained conformal map starts to suffer from occasional numerical issues (e.g., step size going down to numerical zero) as well. Investigation of the numerical aspects of map optimization in high distortion cases is an important direction for future research.

7 CONCLUSION AND FUTURE WORK

This article provides a general path to obtaining seamless parametrizations with a given set of cones. On a conceptual level, the approach is simple: pad a parametrization that maps cut segments to straight lines, with padding determined by solving a linear system. Our algorithm demonstrates that for (almost) any user-specified or automatically computed choice of cones a corresponding global parametrization can be constructed, without introducing additional cones.

A limitation of our approach in its current form is that it does not take into account the holonomy angles on global homology loops (in addition to cone angles), which, for instance, is important for parametrizations following a global guiding field. We expect that by using different forms of cut graph construction, based on given global holonomy angles, many of the ideas will be applicable to such a setting as well; we plan to address this in a separate article.

Other directions of future work include generalization to surfaces with boundaries as well as aligning to tagged feature curves or other prescribed directions on the surface.

In the smooth setting, we have constructively shown the existence of a locally injective, seamless parametrization. In practice, numerical optimization routines bring about additional challenges related to precision limits, which here affects the discrete conformal map computation. A potential path could be the replacement of this initial map computation with a different technique—perhaps exploiting the fact that conformality is not actually required.

APPENDICES

A ILLUSTRATIVE EXAMPLE

We consider the simplest example: a torus with two cones, $k_0 = 2$, $k_1 = 6$, i.e., cone angles π and 3π, shown in Figure 18.

Cut Graph. We cut the surface into a 2-corner and a 6-corner component (cf. Figure 18, top). The cut graph was embedded in the surface in such a way that the total cone curvature contained in each component is compatible with the number of corners in terms of Gauss-Bonnet: a cone with $k_0 = 2$ lies in the 2-corner region, a cone with $k_1 = 6$ in the 6-corner region.

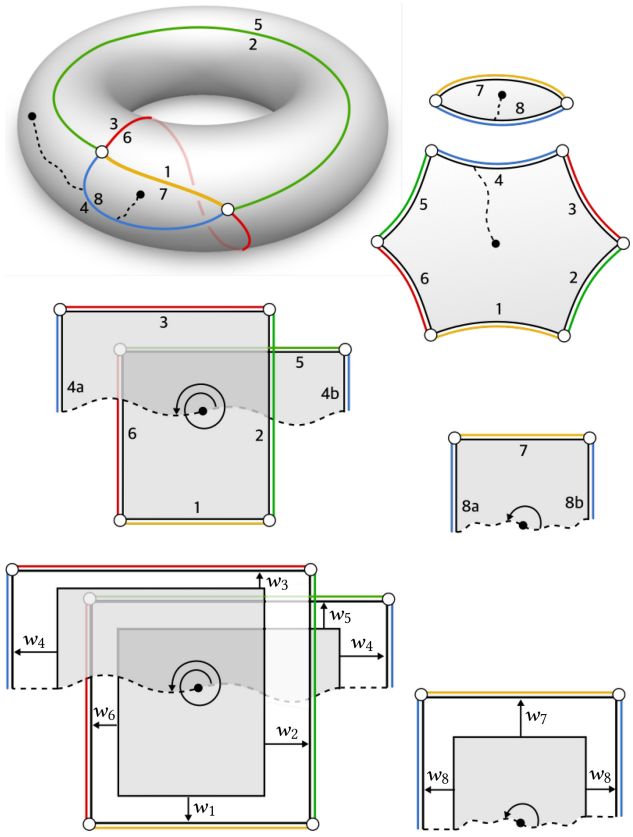


Fig. 18. Top left: genus 1 surface with cut graph consisting of 4 branches (yellow, green, red, blue). The cut graph cuts the surface into two components with 2 and 6 corners, respectively, i.e., with a total of 8 boundary segments (two corresponding to each branch). Top right: schematic depiction of the two components under a cone metric with rectilinear boundary consisting of straight segments (here shown as curved arcs) meeting at right angles. Middle left/right: planar flattening of the two components implied by the metric (after cutting to cones—dashed). The numbering of segments is used to set up the system for padding widths w_i. Bottom left/right: the padded flattening (padding, indicated by arrows, in white).

Cone Metric. We compute a cone metric on each of the two components (e.g., conformal, given by a pointwise scale factor), which is flat everywhere except at the cones, where it has the prescribed curvature. In addition, we require the boundary to be geodesically straight at all boundary points except for the corners, where it forms right angles under the metric.

Metric to Parametrization. If we add cuts connecting all cones to the boundary (indicated with dashed curves), then this cone metric is flat in the interior and corresponds to a global parametrization

of the torus, with two charts (cf. Figure 18, middle, left, and right). The image of each of the two maps is a domain with rectilinear boundary, consisting of straight segments meeting at right angles (excluding the cuts to the cones). As the angle between any two segments is an integer multiple of $\pi/2$, this parametrization is *rotationally* seamless, but it may have a jump in scale across cuts. In particular, two segments corresponding to the same cut graph branch—here (1,7), (2,5), (3,6), and (4,8)—may have different lengths in general.

Equalization by Padding. To obtain a seamless parametrization, we equalize the lengths of identified pairs of segments. This is achieved by adding *padding*; i.e., we extend the parametric domain by shifting straight segments in orthogonal direction (cf. Figure 18, bottom). For each segment i, numbered sequentially around each component, ℓ_i, $i = 1 \dots 8$, is its parametric length. For a segment i, after padding its length becomes $\ell_i + w_{\text{prev}(i)} + w_{\text{next}(i)}$, where $\text{prev}(i)$ and $\text{next}(i)$ are previous and next segment indices around the component, and w_j is the padding width for segment j. Equating the post-padding lengths of all four pairs of identified segments yields the following four equations in this example:

$$\ell_1 + w_2 + w_6 = \ell_7 + 2w_8, \quad \ell_2 + w_1 + w_3 = \ell_5 + w_4 + w_6,$$
$$\ell_3 + w_2 + w_4 = \ell_6 + w_1 + w_5, \quad \ell_4 + w_3 + w_5 = \ell_8 + 2w_7,$$

where ℓ_i are the known segment lengths, w_i are the unknown padding widths. The matrix of this equation system has the form

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & -2 \\ 1 & 0 & 1 & -1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & -2 & 0 \end{bmatrix},$$

and the right-hand side is $\boldsymbol{b} = [\ell_7 - \ell_1,\ \ell_5 - \ell_2,\ \ell_6 - \ell_3,\ \ell_8 - \ell_4]^T$, i.e., parametric length mismatches of identified segments. To *equalize* segments lengths, we need to find a solution of the system $A\boldsymbol{w} = \boldsymbol{b}$, where $\boldsymbol{w}$ is the vector of padding widths, such that $\boldsymbol{w} \geq 0$. This non-negativity condition is important to guarantee that the domain does not degenerate through padding. Observe that A has full (row) rank, which ensures that the system has a (possibly non-unique) solution. Observe further that $A\mathbf{1} = 0$ in this case, i.e., after computing an arbitrary solution, we can obtain a non-negative solution by adding a sufficiently large constant. More generally, note that A is determined solely by the choice of the cut graph combinatorics. For instance, without the blue or without the yellow branch, the cut graph (cutting the surface to a single topological disk in these cases) would yield a system that does *not* have a non-negative solution $\boldsymbol{w}$ for every possible $\boldsymbol{b}$.

Seamless Parametrization. Once the padded domain is obtained, we remap the original image onto this domain. This is done by stretching outwards thin strips running along the segments to cover the added space in the rectangular regions padded onto the domain, yielding a seamless global parametrization.

B PROOFS OF EQUALIZABILITY

B.1 Genus 3+

This proof is constructive, yet it is only intended to prove feasibility. In practice, a simple linear system solve can be used to obtain a solution instead (cf. Section 5.3), while our theorem ensures that this linear system solve always succeeds.

Non-Fourfold Case. The hole chain cut graph G, together with the extra path, cuts M into two components M'_0 and M'_1, neither of which has its number of segments m_k divisible by 4. The boundary of at least one of these components contains a segment of a hole, located between the two parts of an odd-couple (cf. Section 4.1); let this component be M'_1.

Our proof is based on the observation (Lemma B.1) that for each of the domains M'_k, it is possible to attain *arbitrary* target segment lengths $\tilde{\ell}_i$, $i = 0 \dots m_k - 1$, using (possibly negative) padding. Hence, it is, in particular, possible to choose padding widths such that lengths of mated segments match (Lemma B.3).

If there were no T-nodes, then such (possibly negative) equalizing padding widths could easily be transformed into non-negative ones: Adding a sufficiently large constant c to each yields non-negative padding widths while preserving equalization (each segment length increases by $2c$). Cut graphs with T-nodes result in complex segments on the cut, and these are not mated in pairs. Equalization is preserved if we add c to all padding widths *except* those of the two extra path segments—but then these two would remain possibly negative. This requires performing two intermediate modifications:

First, we make both extra path segments' padding widths non-negative (Lemma B.4). This leads to one equalization equation being violated. We then show that, exploiting the presence of an odd-couple, a further modification of padding widths (Lemma B.5) can restore equalization (Proposition B.1).

Remark:. While it may be possible to construct cut graphs without T-nodes for any configuration of genus and cones, and thereby simplify the proof, this would require the consideration of (possibly many) further special cases, with more than two components, depending on genus and cone curvatures. We choose the version leading to a simpler construction algorithm with fewer special cases.

We first establish two auxiliary results for an individual component M'_k, omitting the subscript k. We number segments cyclically along the component's boundary, with numbers from 0 to $m - 1$. Let $B\boldsymbol{w} = \boldsymbol{d}$, with $m \times m$ matrix B, be the *length adjustment* system formed by equations $w_{i-1} + \ell_i + w_{i+1} = \tilde{\ell}_i$ (with index arithmetic done $\bmod m$), for initial lengths ℓ_i and arbitrary target lengths $\tilde{\ell}_i$.

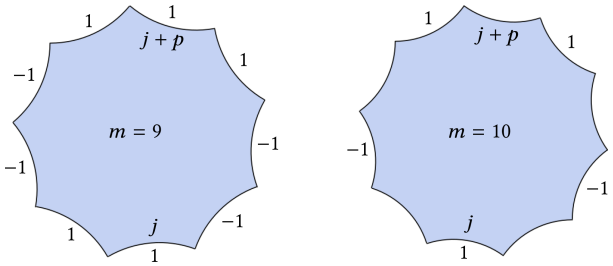
Lemma B.1. *There are padding widths $w_i \in \mathbb{R}$ satisfying $B\boldsymbol{w} = \boldsymbol{d}$.*

Proof. The system matrix B is an $m \times m$ *circulant* matrix, with *associated polynomial* $f(x) = x + x^{m_k - 1} = x(x^{m_k - 2} + 1)$ [Davis 2012]. It is full rank whenever $f(e^{j/m_k 2\pi \mathrm{i}}) \neq 0$, for $j = 0 \dots m_k - 1$, as its determinant is given by the product of these values (f on m^{th} roots of unity). It is straightforward to check that $f(e^{j/m 2\pi \mathrm{i}}) = 0$ (for some j) requires $e^{j(m-2)/m_k 2\pi \mathrm{i}} = -1$. This is equivalent to $2j(m-2)/m$ being odd, which in turn requires $4j/m$ to be odd, i.e., m must be a multiple of 4, contradicting the assumption $m \bmod 4 \neq 0$. □

Lemma B.2. *For an arbitrary choice of index $0 \leq j < m$ and an odd number p, there are padding widths $w_i \in \mathbb{R}$ with $w_j = 1$ such*

that all equations of $B\boldsymbol{w} = \mathbf{0}$, except the $(j + p \mod m)$-th equation, are satisfied.

Proof. Choose $w_{j+p+1}, w_{j+p+3}, w_{j+p+5}, \ldots, w_{j+p-1}$ alternatingly as 1 and -1 (where index summation is done mod m). Note that if m is even, this sequence contains every other padding width, if m is odd, it contains every padding width, as illustrated below.



In both cases, due to $m \bmod 4 \neq 0$, this sequence is of odd length. Thus, $w_{j+p+1} = w_{j+p-1}$. As p is odd, w_j is part of the sequence; we choose the alternating sign such that $w_j = 1$. As each segment $i \neq j + p$ has its previous and next segments either both not padded or padded with alternating signs, its length is not changed, i.e., padded length $\ell'_i = \ell_i$, but $\ell'_{j+p} = \ell_{j+p} \pm 2$. □

Combining the result of Lemma B.1 for both components yields the following lemma.

Lemma B.3. *The equalization system $A\boldsymbol{w} = \boldsymbol{b}$ induced by a hole chain cut graph with admissible extra path has a (possibly negative) solution $\boldsymbol{w}$.*

Proof. Choose the target length $\tilde{\ell}_i = 1$ for each simple segment i (consisting of one boundary curve), and $\tilde{\ell}_i = r$ for each complex segment consisting of r boundary curves (with $r - 1$ flat joints). Note that for our hole chain cut graph $r = 2$ or $r = 3$ (when $m_k = 2$ for some k). According to Lemma B.1, there are padding widths $\boldsymbol{w}_0$ for M'_0 and $\boldsymbol{w}_1$ for M'_1 such that padded lengths $\ell'_i = \tilde{\ell}_i$, and because lengths $\tilde{\ell}_i$ match for mated segments, these padding widths $\boldsymbol{w} = [\boldsymbol{w}_0, \boldsymbol{w}_1]$ are equalizing, thus $A\boldsymbol{w} = \boldsymbol{b}$. □

We now show that this initial, possibly negative solution can, in multiple steps, be transformed into a non-negative solution.

Lemma B.4. *Consider the equalization system $A\boldsymbol{w} = \boldsymbol{b}$ induced by a hole chain cut graph with admissible extra path. Let a, b be the segments of the extra path, q an arbitrary hole segment of M'_1. There are padding widths $\bar{\boldsymbol{w}}$ with $\bar{w}_a = \bar{w}_b = 0$ such that all equalization equations of $A\bar{\boldsymbol{w}} = \boldsymbol{b}$ except for the one containing ℓ_q are satisfied.*

Proof. Lemma B.3 yields padding widths satisfying $A\boldsymbol{w} = \boldsymbol{b}$, but possibly with $w_a \neq 0$ or $w_b \neq 0$. Let $w'_i = w_i$, for all i, except for $w'_a = 0$, $w'_b = w_b + w_a$ (i.e., we move all extra path padding to one side of the path). The equation $A\boldsymbol{w}' = \boldsymbol{b}$ still holds, because the padding widths of segments a and b do not appear individually, but only as sum in these Equations (5), and $w'_a + w'_b = w_a + w_b$.

Observe that due to b being an extra path and q a hole segment index (and the extra path being connected to a hole, cf. Definition 4.2), at least one of the two (cw or ccw) cyclic index distances along the boundary of M'_1 is odd, thus there is an odd number p such that $(b + p \mod m_1) = q$. We now apply Lemma B.2

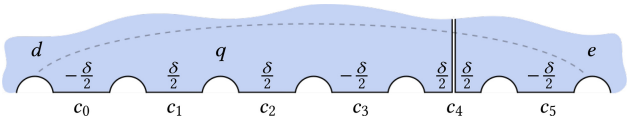


Fig. 19. Illustration of a hole segment q between two segments of an odd-couple d–e (here with 5 hole segments between them). At c_4 an exemplary extra path connection to the hole chain is depicted.

to the component M'_1 to make the padding width of b zero. For the choice $j = b$ and the odd number p, this lemma yields padding widths w''_i for M'_1 (for each segment i of M'_0, we set $w''_i = 0$) with $w''_b = 1$, which leave all segment lengths unchanged except for that of segment q. We obtain padding widths satisfying the lemma as $\bar{w}_i = w'_i - w'_b w''_i$. □

Now, we show that a further modification allows us to find padding width that also satisfies the equation containing ℓ_q. This is the only step that requires using the assumption that the hole chain cut graph contains an odd-couple (cf. Section 4.1).

Lemma B.5. *Suppose the equalization system $A\boldsymbol{w} = \boldsymbol{b}$ is induced by a hole chain cut graph with admissible extra path and an odd-couple. Let q be an arbitrary hole segment of M'_1 between the mated segments d and e of the odd-couple in the hole chain. Then for any $\delta \in \mathbb{R}$ there are padding widths $\boldsymbol{w}$ with $w_a = w_b = 0$ such that the padded length $\ell'_q = \ell_q + \delta$, while all other lengths remain unchanged or are changed by the same amount for each group of mated segments, i.e., $A\boldsymbol{w} = \mathbf{0}$ except for the equation including ℓ_q.*

Proof. Let $d_0, \ldots, d_r$ be the sequence of connectors of the hole chain between an odd-couple pair of holes h_d, h_e. Note that r is odd. Each connector has two sides; let $c_0, \ldots, c_r$ be the sides on that side of the hole chain where the hole segment q is located. Then there are c_i, c_{i+1} such that segment q lies between them (Figure 19).

Let $v_{c_j} = -1^{i-j}\frac{1}{2}\delta$, $j = 0, \ldots, i$, and $v_{c_j} = -1^{i-j+1}\frac{1}{2}\delta$, $j = i+1, \ldots, r$. Each connector side c_j corresponds to one segment s_{c_j} or, in case the extra path connects to such a side and splits it, two segments s_{c_j} and $\bar{s}_{c_j}$ (for example, c_4 in the figure below). Let $w_{s_j} = v_{c_j}$ (and in case of a connector split by the extra path also $w_{\bar{s}_j} = v_{c_j}$) for each c_j, and zero for all other segments not involved in connector sides $c_0, \ldots, c_r$ (thus in particular the extra path segments a, b). With these padding widths, we have $\ell'_q = \ell_q + \delta$, the lengths of mated segments d and e are either both increased or both decreased by the same amount $\frac{1}{2}\delta$. All other hole segments between d and e have their previous and next connector segment padded with opposite signs, preserving their lengths $\ell'_i = \ell_i$—unless the extra path is connected to one of these holes: suppose it is connected to the hole between c_j and c_{j+1}, then there are two hole segments (separated by the extra path) between c_j and c_{j+1}, and one's length is increased by $\frac{1}{2}\delta$, the other's decreased by $\frac{1}{2}\delta$; as they (due to the T-node at the extra path) are mated in combination, these values cancel in the equalization equations. □

Finally, we obtain a non-negative solution of the equalization system, as required.

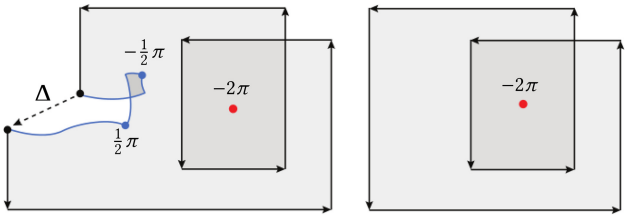


Fig. 20. Left: boundary $\partial F(M^c)$ (black) laid out in the plane after cutting to cones (blue). Red indicates a cone with $k_i = 8$, i.e., curvature $\hat{\Theta}_i = -2\pi$ (parametric angle 4π) for which a cut is superfluous. Right: The segment gap Δ vanishes if all cones are fourfold, thus $\partial F(M^c)$ is a rectilinear polygon.

Proposition B.1. *The equalization system $A\boldsymbol{w} = \boldsymbol{b}$ induced by a hole chain cut graph with admissible extra path and an odd-couple, has a non-negative solution $\boldsymbol{w}$.*

Proof. Let q be a hole segment of M_1' that lies between two segments of an odd-couple. For this q, Lemma B.4 asserts there are padding widths $\bar{\boldsymbol{w}}$ with $\bar{w}_a = \bar{w}_b = 0$, satisfying all but one equalization equation. Let δ be the error, i.e., the difference between the padded length ℓ_q' and the padded length of q's mate(s). For these q and δ, Lemma B.5 yields padding widths $\hat{w}_i$ such that $w_i' = \bar{w}_i + \hat{w}_i$ satisfy all equations $A\boldsymbol{w}' = \boldsymbol{b}$, because with this addition, the padded length ℓ_q' of segment q is adjusted to cancel the error δ, while equalization of all other mated segments is preserved, and $w_a' = w_b' = 0$. Now non-negative equalizing padding widths can be derived: Let $\mathbf{1}'$ be the vector of ones, except for two zeroes at entries a and b. Then $A\mathbf{1}' = \mathbf{0}$, because each Equation (3)/(5), thus each row of A, contains *two* (not necessarily distinct) padding width variables with positive sign and *two* padding width variables with negative sign—and possibly further entries $-w_a, -w_b$, which, however, are zero in $\mathbf{1}'$. Let $\lambda = \min w_i'$. Then padding widths $\boldsymbol{w} = \boldsymbol{w}' - \lambda\mathbf{1}'$ are non-negative and, due to $A\boldsymbol{w} = A\boldsymbol{w}' - \lambda A\mathbf{1}' = \boldsymbol{b}$, equalizing. □

Fourfold Case. In this case, the cut graph contains no extra cut path, and we have a single component M' with the number of segments m divisible by 4. As shown in the proof of Lemma B.1, system matrix B does not have full rank in this case. Its upper left $(m-2)\times(m-2)$-submatrix, however, has full rank (as it is a tridiagonal Toeplitz matrix), thus B^-, which is B with the last two rows removed, is a (rectangular) matrix with full row rank. This implies we can obtain padding widths $\boldsymbol{w}$ with $B^-\boldsymbol{w} = \mathbf{1} - \boldsymbol{\ell}^-$, i.e., they bring all segments *but the last two* to unit length.

In the case of fourfold cones, the transitions across the cut graph extension T (cf. Section 3.3) are rotations by a multiple of 2π (= identity); as a consequence, the cut extension T can actually be omitted. This implies that the boundary of the flattening $F(M')$ is formed exclusively by the segments and is entirely rectilinear, as illustrated in Figure 20 (right). Without the loss of generality, we assume that all even-index segments are laid out horizontally and all odd-index segments vertically. Counterclockwise around the boundary $\partial F(M')$, horizontal segments alternate between positive and negative u-directions, and vertical segments alternate between positive and negative v-direction. The fact that $\partial F(M')$ is

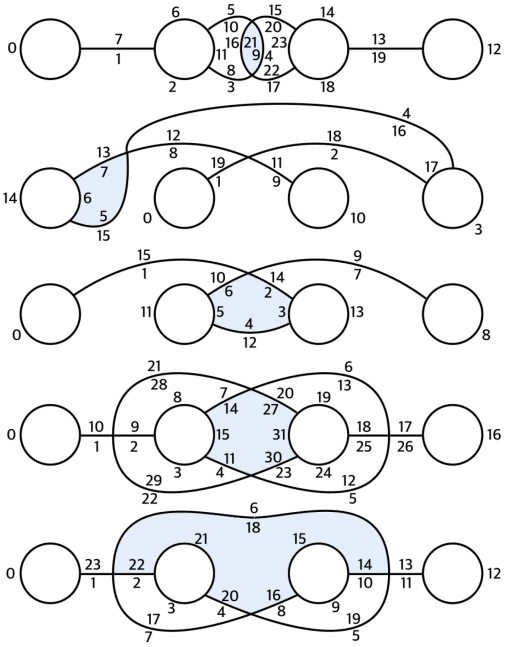


Fig. 21. Special cut graph patterns to be used to guarantee equalizability for genus 2 surfaces, depending on whether a subset of cones compatible with a region (shaded) with 2, 3, 5, 6, or 7 corners is present.

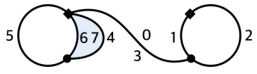
(and after padding remains) a *closed* polygon then implies

$$\sum_{i=0,\dots,m/2-1} -1^i \ell_{2i} = 0, \qquad \sum_{i=0,\dots,m/2-1} -1^i \ell_{2i+1} = 0.$$

This, in turn, implies that if all even/odd segments but one have unit length, the remaining one has unit length as well. Hence, the last two conditions of B are, in the fourfold case, satisfied automatically if all other conditions are satisfied. We conclude that $B^-\boldsymbol{w} = \mathbf{1} - \boldsymbol{\ell}^-$ implies $B\boldsymbol{w} = \mathbf{1} - \boldsymbol{\ell}$. The vector $\boldsymbol{w}$ may contain negative values, but we can add an arbitrary constant shift $\boldsymbol{w}^* = \boldsymbol{w} + \lambda\mathbf{1}$, because $B\boldsymbol{w}^* = \mu\mathbf{1} - \boldsymbol{\ell}$ (with $\mu = (2\lambda + 1)$), leading to $A\boldsymbol{w}^* = \boldsymbol{b}$ and $\boldsymbol{w}^* \geq 0$ for a sufficiently large λ. □

B.2 Genus 1

The equalization system for the genus 1 cut graph pattern is



$$\begin{aligned} w_1 + w_5 - w_2 - w_4 &= \ell_3 - \ell_0 \\ w_6 + w_6 - w_3 - w_5 &= \ell_4 - \ell_7 \\ w_0 + w_2 - w_7 - w_7 &= \ell_6 - \ell_1 \\ w_1 + w_3 - w_0 - w_4 &= \ell_5 - \ell_2 \end{aligned}$$

One can easily verify that the system matrix has full row rank, and that it has positive vectors (e.g., $\mathbf{1}$) in its kernel, thus has a non-negative solution for any righthand side.

B.3 Genus 2

The five different cut graph patterns covering all possible cone choices for genus 2 are depicted in Figure 21. One can easily verify explicitly that their equalization system matrices all have full row rank, and that they have positive vectors (e.g., $\mathbf{1}$) in their kernel, thus have non-negative solutions for any righthand side.

2 corners:

$$w_1 + w_7 - w_{13} - w_{19} = \ell_{12} - \ell_0$$
$$w_1 + w_3 - w_{17} - w_{19} = \ell_{18} - \ell_2$$
$$w_5 + w_7 - w_{13} - w_{15} = \ell_{14} - \ell_6$$
$$w_8 + w_{10} - w_{20} - w_{22} = \ell_{23} - \ell_{11}$$
$$w_0 + w_2 - w_0 - w_6 = \ell_7 - \ell_1$$
$$w_2 + w_{17} - w_{11} - w_{16} = \ell_8 - \ell_3$$
$$w_{20} + w_{22} - w_{21} - w_{21} = \ell_9 - \ell_4$$
$$w_6 + w_{15} - w_{11} - w_{16} = \ell_{10} - \ell_5$$
$$w_{12} + w_{14} - w_{12} - w_{18} = \ell_{19} - \ell_{13}$$
$$w_5 + w_{14} - w_4 - w_{23} = \ell_{20} - \ell_{15}$$
$$w_8 + w_{10} - w_9 - w_9 = \ell_{21} - \ell_{16}$$
$$w_3 + w_{18} - w_4 - w_{23} = \ell_{22} - \ell_{17}$$

3 corners:

$$w_1 + w_{19} - w_9 - w_{11} = \ell_{10} - \ell_0$$
$$w_5 + w_7 - w_{16} - w_{18} = \ell_{17} - \ell_6$$
$$w_2 + w_4 - w_{13} - w_{15} = \ell_{14} - \ell_3$$
$$w_0 + w_9 - w_0 - w_8 = \ell_{19} - \ell_1$$
$$w_3 + w_{11} - w_{12} - w_{17} = \ell_{18} - \ell_2$$
$$w_3 + w_{13} - w_{12} - w_{17} = \ell_{16} - \ell_4$$
$$w_6 + w_7 - w_8 - w_{14} = \ell_{15} - \ell_5$$
$$w_5 + w_6 - w_4 - w_{14} = \ell_{13} - \ell_7$$
$$w_{15} + w_{19} - w_{16} - w_{18} = \ell_{12} - \ell_8$$
$$w_1 + w_{10} - w_2 - w_{10} = \ell_{11} - \ell_9$$

5 corners:

$$w_1 + w_{15} - w_7 - w_9 = \ell_8 - \ell_0$$
$$w_2 + w_4 - w_4 - w_6 = \ell_5 - \ell_3$$
$$w_{10} + w_{12} - w_{12} - w_{14} = \ell_{13} - \ell_{11}$$
$$w_0 + w_{10} - w_0 - w_9 = \ell_{15} - \ell_1$$
$$w_3 + w_6 - w_7 - w_{13} = \ell_{14} - \ell_2$$
$$w_3 + w_5 - w_{11} - w_{13} = \ell_{12} - \ell_4$$
$$w_2 + w_5 - w_1 - w_{11} = \ell_{10} - \ell_6$$
$$w_8 + w_{14} - w_8 - w_{15} = \ell_9 - \ell_7$$

7 corners:

$$w_1 + w_{23} - w_{11} - w_{13} = \ell_{12} - \ell_0$$
$$w_2 + w_4 - w_8 - w_{10} = \ell_9 - \ell_3$$
$$w_{14} + w_{16} - w_{20} - w_{22} = \ell_{21} - \ell_{15}$$
$$w_0 + w_7 - w_0 - w_6 = \ell_{23} - \ell_1$$
$$w_3 + w_{17} - w_{18} - w_{21} = \ell_{22} - \ell_2$$
$$w_3 + w_{17} - w_{16} - w_{21} = \ell_{20} - \ell_4$$
$$w_7 + w_{11} - w_8 - w_{10} = \ell_{19} - \ell_5$$
$$w_{13} + w_{23} - w_{14} - w_{22} = \ell_{18} - \ell_6$$
$$w_1 + w_5 - w_2 - w_4 = \ell_{17} - \ell_7$$
$$w_9 + w_{19} - w_{15} - w_{20} = \ell_{16} - \ell_8$$
$$w_9 + w_{19} - w_{15} - w_{18} = \ell_{14} - \ell_{10}$$
$$w_5 + w_{12} - w_6 - w_{12} = \ell_{13} - \ell_{11}$$

6 corners:

$$w_1 + w_{10} - w_{17} - w_{26} = \ell_{16} - \ell_0$$
$$w_2 + w_4 - w_{23} - w_{25} = \ell_{24} - \ell_3$$
$$w_7 + w_9 - w_{18} - w_{20} = \ell_{19} - \ell_8$$
$$w_{11} + w_{14} - w_{27} - w_{30} = \ell_{31} - \ell_{15}$$
$$w_0 + w_{22} - w_0 - w_{21} = \ell_{10} - \ell_1$$
$$w_3 + w_{29} - w_8 - w_{28} = \ell_9 - \ell_2$$
$$w_3 + w_{29} - w_{15} - w_{30} = \ell_{11} - \ell_4$$
$$w_{22} + w_{26} - w_{23} - w_{25} = \ell_{12} - \ell_5$$
$$w_{17} + w_{21} - w_{18} - w_{20} = \ell_{13} - \ell_6$$
$$w_8 + w_{28} - w_{15} - w_{27} = \ell_{14} - \ell_7$$
$$w_6 + w_{16} - w_5 - w_{16} = \ell_{26} - \ell_{17}$$
$$w_{13} + w_{19} - w_{12} - w_{24} = \ell_{25} - \ell_{18}$$
$$w_{13} + w_{19} - w_{14} - w_{31} = \ell_{27} - \ell_{20}$$
$$w_6 + w_{10} - w_7 - w_9 = \ell_{28} - \ell_{21}$$
$$w_1 + w_5 - w_2 - w_4 = \ell_{29} - \ell_{22}$$
$$w_{12} + w_{24} - w_{11} - w_{31} = \ell_{30} - \ell_{23}$$

C PROOF OF CONE METRIC EXISTENCE

[Troyanov 1991] presents a general proof of cone metric existence on closed surfaces. It "extends [...] to surfaces with (piecewise geodesic) boundary," but the extension is not spelled out. Cherrier [1984] focuses on the case with boundary, but does not specifically consider the relevant delta distributions of curvature. We provide a proof tailored to our setting.

Let M' be one of the disk-topology connected components of the cut surface (we will drop the index of the component in the following). Consider the expansion M'_{exp} of M', obtained by joining a copy of the geodesic disk of size ϵ_p in M centered at $\pi(p)$, to each boundary point of M'. Multiple $p \in \partial M'$ corresponding to the same $\pi(p)$ get separate copies of the disk centered at $\pi(p)$ and ϵ_p is chosen sufficiently small for each p so that M'_{exp} still has disk topology.

To simplify the exposition, we assume that on the surface M the branches of the cut form right angles—the proof can be extended to arbitrary angles, as long as the curves are transversal, but requires a more complex solution ϕ_1 below, with additional cones at the corners, as explained in more detail in Bunin [2008].

Consider a conformal map f from M'_{exp} to the plane (e.g., to a disk). As M'_{exp} has disk-topology, such a map exists. As M' is in the interior of M'_{exp} the map is conformal at the points of the boundary $\partial M'$. The conformal scale factor $|f'|$, where f' is the complex derivative of the map expressed in local complex coordinates on the tangent plane, defines the conformal metric on $\text{Int}(f(M'_{\text{exp}}))$, in particular, on all of $f(M') = M''$ including the boundary. Let γ_i, $i = 1 \dots m$, be the curves of the boundary of M''; these curves are smooth, as the boundary of M' is smooth, and meet at right angles.

We now construct on M'' a metric with the desired properties; then the metric on M' is obtained by a pullback through f. As M'' is flat, the equation for the metric in the interior points x of M simplifies to

$$\Delta\phi = \sum_j \hat{\Theta}_j \delta(f(p_j) - x),$$

where $\hat{\Theta}_j$ is the target curvature at cone $c_j = (p_j, \hat{\Theta}_j)$. If the geodesic curvature at non-corner boundary points is given by a smooth function κ, then we have the Neumann boundary condition

$$\frac{\partial\phi}{\partial n} = -\kappa,$$

which needs to be satisfied to obtain straight boundary edges in the final metric. Note that κ may be discontinuous at the corner points but it is still in L_2. We can find a particular solution u_1 satisfying the Poisson equation on M'' without boundary conditions directly as $\phi_1 = \sum_j \hat{\Theta}_j \ln(|z - f(p_j)|)$ with singularities at $f(p_j)$.

Then, we solve the Laplace equation $\Delta\phi_2 = 0$, for ϕ_2 with smooth Neumann conditions $\partial\phi_2/\partial n = \kappa - \partial\phi_1/\partial n$. For this problem to have a solution, the Neumann boundary condition needs to integrate to zero over the boundary. Observe that because the domain M'' is flat, the integral of the geodesic curvature κ over the boundary, with the sum of corner angles $n\frac{\pi}{2}$ added, must be 2π, i.e., $\int_{\partial\Omega} \kappa ds = 2\pi - n\frac{\pi}{2}$. In addition, $\int_{\partial\Omega} \partial\phi_2/\partial n ds = \sum_j \hat{\Theta}_j$, by the Gauss theorem. Finally, note that by the cut graph admissibility assumption on the number of corners, $2\pi - n\frac{\pi}{2} - \sum_j \hat{\Theta}_j = 0$, i.e., the integral condition for the Neumann problem is satisfied. Therefore, the problem has a unique, up to a constant, solution. This solution is in H^2 (and, by Sobolev Lemma, C^0 up to the boundary) for domains with piecewise smooth boundary and convex corners between curves (cf. Grisvard [1985], p. 174). The sum $\phi = \phi_1 + \phi_2$ satisfies the Poisson equation and boundary conditions. The metric ϕ is nonsingular at the boundary, therefore it is conformal, and the angles between boundary curves are preserved. We conclude that the pullback of this metric to M' is the needed metric.

D MAP PADDING

As laid out in Section 3.4, map padding consists of the application of stretch maps to rectangular regions, and lateral shifts within these. To define these precisely, we, w.l.o.g., consider the case of a horizontal segment s_j (aligned with the u-axis in (u, v) coordinates) to be padded by w_j in positive v direction, as illustrated in Figure 5—the other cases (negative v, and positive/negative u) are handled analogously.

In the case of a segment split by T, we assume that T (which can be chosen freely) meets the segment at a right angle with a straight cut in the parametric domain. Then both parts can be treated separately using the following operations without special case handling—except for the same rectangle thickness being used for both parts.

Stretch. Let τ_j be the thickness (here: the height) of rectangle R_j, and $(u_{j\min}, v_{j\min})$ the coordinates of the lower left corner of R_j. The map g_i applied to the strip to perform the stretching is a simple one-dimensional scaling by factor $o_j = \frac{w_j+\tau_j}{\tau_j}$:

$$g_j : (u, v) \mapsto (u, v_{j\min} + o_j(v - v_{j\min})). \tag{7}$$

Shift. We apply a deformation (lateral shift) within a rectangle R_j^p that leads to a (piecewise) constant speed parametrization of the segment s_j. We use a simple blend (linear in v) between the map $\phi_j : [u_{j\min}, u_{j\max}] \to [u_{j\min}, u_{j\max}]$ that reparametrizes segment s_j to (piecewise) constant speed (applied at the top of the strip) and the identity map $u \mapsto u$ (applied at the bottom):

$$r_j : (u, v) \mapsto (t\phi_j(u) + (1 - t)u, v), \tag{8}$$

where $t = (v - v_{j\min})/\tau_j$ is the normalized relative v-coordinate within R_j^p. ϕ_j is a constant speed reparametrizaton for simple segments. For complex segments it is with piecewise constant speed, constant per boundary curve the segment consists of, such that the lengths of these boundary curves after reparametrization are in the same ratio as the padded lengths of their mates.

One easily verifies that r_j is injective: the determinant of its Jacobian is $\det J(u, v) = (\partial\phi_j/\partial u(u) - 1)\, t + 1$, and due to $0 \le t \le 1$ and $\partial\phi_j/\partial u(u) > 0$ (as the scaled arc-length reparametrization is non-degenerate and orientation preserving) it is always positive.

Proof of Proposition 3.3. F is rotationally seamless, in particular locally injective and continuous (on M^c). If $F^{p,m}$ is continuous, then so is $f^{(p,m+1)}$, because g_{m+1} is continuous and it is identity on the interface between S_{m+1} and the rest of M^c. If $F^{s,m}$ is continuous, then so is $F^{(s,m+1)}$, because r_{m+1} is continuous and it is identity on the interface between S_{m+1} and the rest of M^c. It follows that F^s is continuous. Analogously, as g_j and r_j are injective, local injectivity is preserved for F^s. Both types of maps, g_j and r_j, preserve the straightness and the orientation of all segments and therefore the pairwise angles between them, thus F^s is rotationally seamless like F. As angles between boundary curve images are not affected, cone angles are preserved as well. Each boundary curve segment that s_j consists of is parametrized with constant speed by $F^{s,j}$ by construction. As s_k with $k \neq j$ is identity on s_j (more precisely: that part of s_j contained in R_k and thus potentially affected by s_k), $F^s(s_j) = F^{s,j}(s_j)$. □

ACKNOWLEDGMENTS

Models shown in Figures 15 and 16: heptoroid generated by "Sculpture Generator 1" by Carlo H. Séquin, UC Berkeley; Thai statue courtesy of Stanford 3D Scanning Repository based on data from XYZ RGB Inc.; botijo courtesy of Image-based 3D Models Archive, Télécom Paris; other models from the AIM@Shape repository.

REFERENCES

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

Noam Aigerman and Yaron Lipman. 2016. Hyperbolic orbifold tutte embeddings. *ACM Trans. Graph.* 35, 6 (2016), 217:1–217:14.

Mirela Ben-Chen, Craig Gotsman, and Guy Bunin. 2008. Conformal flattening by curvature prescription and metric scaling. *Comput. Graph. Forum* 27, 2 (2008), 449–458.

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

David Bommes, Bruno Lévy, Nico Pietroni, Enrico Puppo, Claudio Silva, Marco Tarini, and Denis Zorin. 2013b. Quad-mesh generation and processing: A survey. In *Computer Graphics Forum*. Wiley.

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

Alon Bright, Edward Chien, and Ofir Weber. 2017. Harmonic global parametrization with rational holonomy. *ACM Trans. Graph.* 36, 4 (2017), 89:1–89:15.

Guy Bunin. 2008. A continuum theory for unstructured mesh generation in two dimensions. *Comput. Aid. Geom. 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. Dual strip weaving: Interactive design of quad layouts using elastica strips. *ACM Trans. Graph.* 33, 6 (2014), 183:1–183:10.

Marcel Campen and Denis Zorin. 2017. Similarity maps and field-guided t-splines: A perfect couple. *ACM Trans. Graph.* 36, 4 (2017).

Pascal Cherrier. 1984. Problèmes de neumann non linéaires sur les variétés riemanniennes. *J. Function. Anal.* 57, 2 (1984), 154–206.

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.

Keenan Crane, Mathieu Desbrun, and Peter Schröder. 2010. Trivial connections on discrete surfaces. *Comput. Graph. Forum* 29, 5 (2010), 1525–1533.

Philip J Davis. 2012. *Circulant Matrices*. American Mathematical Society.

Tamal K. Dey, Fengtao Fan, and Yusu Wang. 2013. An efficient computation of handle and tunnel loops via Reeb graphs. *ACM Trans. Graph.* 32, 4 (2013), 32:1–32:10.

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.

Jeff Erickson and Kim Whittlesey. 2005. Greedy optimal homotopy and homology generators. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*. 1038–1046.

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.

M. S. Floater and K. Hormann. 2005. Surface parameterization: A tutorial and survey. In *Advances in Multiresolution for Geometric Modelling*, Neil A. Dodgson, Michael S. Floater, and Malcolm A. Sabin (Eds.).

Michael S. Floater. 1997. Parametrization and smooth approximation of surface triangulations. *Comput. Aid. Geom. Design* 14, 3 (1997), 231–250.

Xiao-Ming Fu, Yang Liu, and Baining Guo. 2015. Computing locally injective mappings by advanced MIPS. *ACM Trans. Graph.* 34, 4 (2015), 71:1–71:12.

Steven J. Gortler, Craig Gotsman, and Dylan Thurston. 2006. Discrete one-forms on meshes and applications to 3D mesh parameterization. *Comput. Aid. Geom. Design* 23, 2 (2006), 83–112.

Pierre Grisvard. 1985. Elliptic problems in nonsmooth domains, vol. 24 of *Monographs and Studies in Mathematics*. Pitman.

Xianfeng Gu, Ren Guo, Feng Luo, Jian Sun, and Tianqi Wu. 2018a. A discrete uniformization theorem for polyhedral surfaces II. *J. Diff. Geom.* 109, 3 (2018), 431–466.

Xianfeng Gu, Feng Luo, Jian Sun, and Tianqi Wu. 2018b. A discrete uniformization theorem for polyhedral surfaces. *J. Diff. Geom.* 109, 2 (2018).

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

Eden Fedida Hefetz, Edward Chien, and Ofir Weber. 2019. A subspace method for fast locally injective harmonic mapping. In *Computer Graphics Forum*, Vol. 38. Wiley, 105–119.

K. Hormann and G. Greiner. 2000. MIPS: An efficient global parametrization method. In *Curve and Surface Design: Saint-Malo 1999*. Vanderbilt University Press, 153–162.

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. *Comput. Graph. Forum* 26, 3 (2007), 375–384.

Liliya Kharevych, Boris Springborn, and Peter Schröder. 2006. Discrete conformal mappings via circle patterns. *ACM Trans. Graph.* 25, 2 (2006), 412–438.

Shahar Z. Kovalsky, Meirav Galun, and Yaron Lipman. 2016. Accelerated quadratic proxy for geometric optimization. *ACM Trans. Graph.* 35, 4 (2016), 134:1–134:11.

Feng Luo. 2004. Combinatorial yamabe flow on surfaces. *Commun. Contemp. Math.* 6, 05 (2004), 765–780.

Max Lyon, Marcel Campen, David Bommes, and Leif Kobbelt. 2019. Parametrization quantization with free boundaries for trimmed quad meshing. *ACM Trans. Graph.* 38, 4 (2019), 51:1–51:14.

Manish Mandad and Marcel Campen. 2019. Exact constraint satisfaction for truly seamless parametrization. In *Computer Graphics Forum*, Vol. 38. Wiley, 135–145.

Giorgio Marcias, Nico Pietroni, Daniele Panozzo, Enrico Puppo, and Olga Sorkine-Hornung. 2013. Animation-aware quadrangulation. In *Proceedings of the Symposium on Geometry Processing*. 167–175.

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.
Budirijanto Purnomo, Jonathan D. Cohen, and Subodh Kumar. 2004. Seamless texture atlases. In *Proceedings of the Symposium on Geometry Processing*. 65–74.
Michael Rabinovich, Roi Poranne, Daniele Panozzo, and Olga Sorkine-Hornung. 2017. Scalable locally injective mappings. *ACM Trans. Graph.* 36, 2 (2017), 16:1–16:16.
Nicolas Ray, Vincent Nivoliers, Sylvain Lefebvre, and Bruno Lévy. 2010. Invisible seams. In *Proceedings of the EG Symposium on Rendering*.
Rohan Sawhney and Keenan Crane. 2017. Boundary first flattening. *ACM Trans. Graph.* 37, 1 (2017), 5:1–5:14.
Christian Schüller, Ladislav Kavan, Daniele Panozzo, and Olga Sorkine-Hornung. 2013. Locally injective mappings. *Comput. Graph. Forum* 32, 5 (2013), 125–135.
Anna Shtengel, Roi Poranne, Olga Sorkine-Hornung, Shahar Z. Kovalsky, and Yaron Lipman. 2017. Geometric optimization via composite majorization. *ACM Trans. Graph.* 36, 4 (2017), 38:1–38:11.
Yousuf Soliman, Dejan Slepčev, and Keenan Crane. 2018. Optimal cone singularities for conformal flattening. *ACM Trans. Graph.* 37, 4 (2018), 105:1–105:17.
B. Springborn. 2017. Hyperbolic polyhedra and discrete uniformization. *ArXiv E-prints* arxiv:1707.06848
Boris Springborn, Peter Schröder, and Ulrich Pinkall. 2008. Conformal equivalence of triangle meshes. *ACM Trans. Graph.* 27, 3 (2008).
Yiying Tong, Pierre Alliez, Daniel Cohen-Steiner, and Mathieu Desbrun. 2006. Designing quadrangulations with discrete harmonic forms. In *Proceedings of the Symposium on Geometry Processing*. 201–210.
Marc Troyanov. 1991. Prescribing curvature on compact surfaces with conical singularities. *Trans. Amer. Math. Soc.* 324 (1991), 793–821.
W. T. Tutte. 1963. How to draw a graph. *Proc. Lond. Math. Soc.* 13 (1963), 743–767.
Amir Vaxman, Marcel Campen, Olga Diamanti, Daniele Panozzo, David Bommes, Klaus Hildebrandt, and Mirela Ben-Chen. 2016. Directional field synthesis, design, and processing. *Comput. Graph. Forum* 35, 2 (2016).
Ana-Maria Vintescu, Florent Dupont, Guillaume Lavoué, Pooran Memari, and Julien Tierny. 2017. Conformal factor persistence for fast hierarchical cone extraction. In *Proceedings of the Eurographics Conference (EG'17)*.
Jiaran Zhou, Marcel Campen, Denis Zorin, Changhe Tu, and Claudio T Silva. 2018. Quadrangulation of non-rigid objects using deformation metrics. *Comput. Aid. Geom. Design* 62 (2018), 3–15.
Yufeng Zhu, Robert Bridson, and Danny M. Kaufman. 2018. Blended cured quasi-newton for distortion optimization. *ACM Trans. Graph.* 37, 4 (2018), 40:1–40:14.

Received November 2018; revised April 2019; accepted July 2019