Bounded Distortion Parametrization in the Space of Metrics

Edward Chien
Bar Ilan University, Israel

Zohar Levi
Victoria University of Wellington, New Zealand

Ofir Weber
Bar Ilan University, Israel

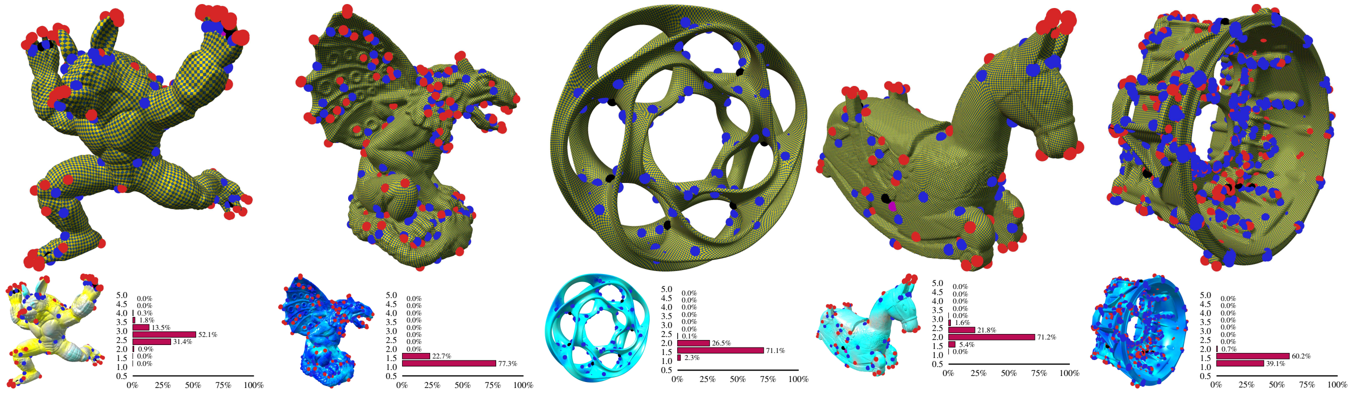


Figure 1: *Robustness test. We picked five models from a recent benchmark [Myles et al. 2014] that several state-of-the-art methods consistently failed to address. The constrained optimization method of [Aigerman et al. 2014] failed due to infeasibility. CETM [Springborn et al. 2008] failed to flatten the metrics due to triangle inequality violation. Lastly, we applied the $ARAP - L_\infty$ [Levi and Zorin 2014] approach which always produces a layout, but the parametrizations were not locally injective. Variant V1 of our method converged to a flat cone metric, resulting in a locally injective parametrization with bounded isometric distortion (as can be seen by the highest values in the histogram and the color visualization). The distortion values plotted are the $\sqrt{\tau}$ values for each triangle (see Section 3.2.1).*

Abstract

We present a framework for global parametrization that utilizes the edge lengths (squared) of the mesh as variables. Given a mesh with arbitrary topology and prescribed cone singularities, we flatten the original metric of the surface under strict bounds on the metric distortion (various types of conformal and isometric measures are supported). Our key observation is that the space of bounded distortion metrics (given any particular bounds) is convex, and a broad range of useful and well-known distortion energies are convex as well. With the addition of nonlinear Gaussian curvature constraints, the parametrization problem is formulated as a constrained optimization problem, and a solution gives a locally injective map. Our method is easy to implement. Sequential convex programming (SCP) is utilized to solve this problem effectively. We demonstrate the flexibility of the method and its uncompromised robustness and compare it to state-of-the-art methods.

Keywords: mesh parametrization, injective maps, bounded distortion

Concepts: •**Computing methodologies** → **Computer graphics; Shape modeling; Mesh models; Mesh geometry models;**

1 Introduction

Parametrization is a classic problem in computer graphics and geometry processing with abundant uses and applications such as remeshing, texture mapping, shape correspondence, rotational symmetry fields-and-patterns design, and compression.

A continuous and (locally) injective map of a surface to the plane exists only for topological disks (possibly with holes). A global parametrization of an orientable surface with arbitrary topology can be obtained by equipping the surface with a metric which is flat everywhere except at isolated cone singularities with nonzero Gaussian curvature. The surface is then cut to a disk, such that the cut passes through all cones and a layout of the cut-to-disk surface can be uniquely defined (up to rigid motion).

Other than altering the topology of the surface, introducing cuts can lead to dramatic reduction in the metric distortion of the map. However, finding a flat metric with prescribed cone angles that has minimal and/or bounded distortion is still highly challenging. In fact, even the simpler task of parametrizing the cut-to-disk surface, such that the induced map is locally injective (regardless of the amount of distortion) is still a hard problem. A wide variety of approaches exist with different success rates but the underlying optimization problems are inherently nonconvex and guaranteeing a valid solution without altering the mesh connectivity or the cone structure [Myles et al. 2014] is elusive.

A key insight in this paper (Sections 4.2.1 & 4.3.1) is that for a simplicial mesh manifold of arbitrary dimension $n \geq 2$, the space of discrete metrics with bounded conformal and isometric distortion is a *convex* set. We focus on the special case of triangular mesh surfaces, for which the bounded distortion space can be characterized by the intersection of multiple (convex) second-order cones.

The majority of existing parametrization methods, use measures of metric distortion that can be characterized in terms of the singular values of the Jacobian J_f of a continuous piecewise linear map f.

SA '16 Technical Papers, December 05 - 08, 2016, Macao
ISBN: 978-1-4503-4514-9/16/12
DOI: http://dx.doi.org/10.1145/2980179.2982426

ACM Reference Format
Chien, E., Levi, Z., Weber, O. 2016. Bounded Distortion Parametrization in the Space of Metrics.
ACM Trans. Graph. 35, 6, Article 215 (November 2016), 16 pages. DOI = 10.1145/2980179.2982426
http://doi.acm.org/10.1145/2980179.2982426.

Based on the singular value decomposition (SVD), it is easy to verify that these singular values, $\sigma_a \geq \sigma_b \geq 0$ are in fact the square roots of the eigenvalues of the so-called metric tensor $M_f = J_f^\top J_f$. Current approaches for mesh parametrization mostly work with the spatial coordinates (the u and v coordinates of the vertices in the parametric domain) as variables. Our point of departure is to propose alternative variables, which allow for better control over the distortion. These variables are the edge lengths squared, which are linear expressions in the entries of the metric tensor M_f, and can be viewed as the discrete metric.

We draw inspiration from the work of [Chen et al. 2013], which linearly blends the edge lengths squared of two meshes in order to perform planar shape interpolation with bounded conformal distortion. To the best of our knowledge, using the metric directly as variables for the sake of mesh parametrization was not considered before. We design an algorithm for computing a bounded distortion global parametrization of an orientable triangle mesh with arbitrary topology (high genus, open/closed) with prescribed cone singularities.

As noted, the finite-dimensional space of bounded distortion discrete metrics on the mesh is shown to be convex in our variables, allowing us to directly control the distortion of our candidate metric. Bounding the distortion is attained through convex second-order cone inequality constraints. A variety of well-known energies that strive to reduce the maximal and/or the average metric distortion, are shown to be convex in our variables. In order to parametrize the mesh, however, additional prescription of the Gaussian curvature is required. The curvature constraint is unfortunately, not convex, but it can be effectively enforced by a simple iterative algorithm.

We apply the sequential convex programming (SCP) method [Dinh and Diehl 2010] to solve the optimization problem. The algorithm is remarkably simple and boils down to iterating two simple steps. First, the nonlinear curvature function is approximated locally by a linear function; then, this linearized curvature constraint is incorporated into the solution of a second-order cone program (SOCP). The success of the approach is justified by the ability to solve the convex subproblems efficiently and accurately with the help of modern interior point solvers [ApS 2015]. Moreover, the nonlinear curvature function is smooth and well-behaved due to the existence of the distortion bounds which prohibit degeneracies and triangle inequality violation.

Once the metric is flattened, the mesh is cut to a disk and laid down in the plane. Flatness at vertices not on the cuts ensures local injectivity. Also, since cutting is not done a priori, the transition along the seams are rigid transformations, and scale (metric) consistency across the two sides of the seam is automatically obtained. The algorithm supports arbitrary cone angle prescriptions, including angles that are multiples of 90 degrees (so-called seamless prescriptions [Myles and Zorin 2012]), though we do not address in this work the problem of cone placement or angle rounding, or the quantization of cone positions in the parameter space [Campen et al. 2015]. Our main focus in this work is on computing a new metric with bounded distortion, given prescription of the Gaussian curvature at all the vertices. Our method can be seen as solving a generalization of the problem addressed by conformal methods [Kharevych et al. 2006; Ben-Chen et al. 2008; Springborn et al. 2008], which aim to parametrize while staying in the more restricted space of conformally equivalent metrics.

2 Previous Work

Due to the abundance of literature on mesh parametrization, we will focus only on the most relevant approaches to ours, and in particular on methods that produce locally injective maps and provide bounds on the induced distortion. For in-depth reviews, we refer the reader to [Floater and Hormann 2005; Hormann et al. 2007] for mesh parametrization and the related topics of quad remeshing [Bommes et al. 2013b] and directional field design [Vaxman et al. 2016].

The most common approach to mesh parametrization is to associate spatial coordinates $(u_i, v_i) \in \mathbb{R}^2$ for each mesh vertex and then solve an optimization problem in which these are variables, such that a certain objective function is minimized and/or additional constraints are satisfied. With these approaches, handling a surface which is not homeomorphic to a disk is done by cutting the mesh *a priori* to a topological disk, before it is mapped to the plane. Discontinuity of the map across the two sides of the cut (sometimes referred to as the seam) is unavoidable, and without special treatment, artifacts arise [Lévy et al. 2002]. Scale consistency between edges of the seam that are split, can be obtained if the transition functions between the two parts of the seam are forced to be rigid motions (e.g. [Kälberer et al. 2007; Bommes et al. 2009; Myles and Zorin 2012]).

Working with spatial coordinates presumably offer some comfort for parametrization since the image of the mapped surface is implicitly forced to lie in $\mathbb{R}^2$. However, that alone does not imply that the metric induced by the planar coordinates is flat. An interior vertex with coordinates $(u_i, v_i) \in \mathbb{R}^2$ is flat, only if the sum of angles of incident triangles is precisely 2π. Alternatively, the flatness condition can be characterized by the local injectivity of the map from the mesh to the plane. Such a piecewise linear map is locally injective if each 1-ring neighborhood of the mesh is mapped bijectively (one-to-one and onto) to the plane. A common misconception is that positive orientation of image triangles (no triangle "flips") ensures local injectivity. While this is a necessary condition, it is not a sufficient one. Loosely speaking, positive orientation only implies positivity of angles, but the 1-ring can wind twice around the center vertex, leading to angle sums of 4π (see Figure 1 in [Weber and Zorin 2014] for an illustration). Methods that address local injectivity while working with spatial coordinates, approach the problem by forcing only the necessary condition. To guarantee local injectivity, they must start from a locally injective map (e.g. by using a barrier method [Schüller et al. 2013]) and perform homotopic changes that preserve the orientation of the triangles.

Bounded distortion convexification. The condition for positive orientation of a triangle in the (u, v) plane, can be expressed as a nonconvex quadratic inequality in the (u_i, v_i) variables. A successful line of methods that were developed in recent years force positive orientation (or more sophisticated distortion bounds) by convexification, that is, the nonconvex constraint is substituted with a more strict convex constraint. [Lipman 2012; Bommes et al. 2013a] use it in the context of surface parametrization, while [Aigerman et al. 2014] use it for bijective maps between surfaces. [Kovalsky et al. 2014] generalizes [Lipman 2012] to volumetric maps. [Poranne and Lipman 2014; Chen and Weber 2015] apply similar ideas in the context of planar shape deformation for the meshless setting. Similarly to our paper, all these methods solve a sequence of convex programs: quadratic (QP), second-order cone (SOCP), or semidefinite (SDP) programming, where at each iteration, the convex subspace is altered according to the solution to the previous iteration. In contrast to ours, these methods require careful initialization of a choice of a local frame per triangle. Improper initialization may lead to an empty convex subspace (in the first iteration). The (convex) problem becomes infeasible, and no solution is available. Alternative approaches approximate the bounded distortion space rather than convexify it [Aigerman and Lipman 2013]. These can be accelerated [Kovalsky et al. 2015] and, in contrast to the convexification approach, have the advantage that they always produce a map (though not necessarily a locally injective one). [Liu et al. 2016] also identifies the risk of complete failure due to infeasibility of the constrained optimization approaches and designed a penalty-type solver to address positive orientation of elements in two-and-three dimensions.

Due to the inherent nonconvexity of the underlying problem, none of the existing methods can guarantee a valid solution in every scenario. Our method is no exception and cannot always succeed. Nonetheless, it deviates from existing methods for computation of bounded distortion maps by several aspects. First, we solve the problem in the space of metrics rather than the space of maps. Somewhat surprisingly, the bounded distortion metrics are shown to be a convex space within the space of metrics. We cast all the nonconvexity aspects of the problem into one place, namely the curvature constraint. Forcing flatness (a 2π angle sum) at non-cut vertices ensures local injectivity, avoiding the injectivity issues faced by use of spatial coordinates. This leads to a robust optimization procedure that frees the user from choosing a "good" initialization (we simply start by evolving the original metric). In contrast to convexification approaches, our convex subproblems are always feasible. A failure of our method can only occur if the optimization cannot flatten the metric completely. Our experiments show that this rarely happens, yet, if a failure occurs, we can always provide a partial recovery. After each iteration, the obtained metric (which may be nonflat) is guaranteed to satisfy the distortion bounds. Upon potential failure to converge to a flat metric, the partially flat metric can be used as an input to any other algorithm. For example, the conformal method of [Springborn et al. 2008] which addresses a similar setting by solving a convex problem.

Angle-based approaches. Methods that use triangle angles as variables are closely related to our metric approach, as the angles of a mesh are directly determined from its edge lengths. In contrast to our problem, curvature constraints are linear (sum of angles around a vertex). Albeit, determining mesh edge lengths from a collection of angles is not always possible. Redundancy occurs as there are twice the number of angles than edges (in a closed mesh). Angle based flattening (ABF) [Sheffer and de Sturler 2001] provides the necessary and sufficient conditions for the successful recovery of a flat metric from angles. Namely, the angles should all be positive (linear inequality constraint), they should sum to π on each triangle (linear equality), and similarly to our problem, needs to satisfy a single nonlinear equality constraint per vertex (a product of sines in ABF). In spite of being a nonconvex problem, ABF is quite robust and effective in approximating conformal maps, yet direct control over scale in terms of angles requires an additional integral nonlinear condition. [Bobenko et al. 2015] notes a variational principle for discrete conformality based on angles, and performs a convex unconstrained energy minimization to solve for them.

Metric-based approaches. Better control over scale can be obtained by using different variables. [Kharevych et al. 2006] uses triangle circumcircle radii, while [Ben-Chen et al. 2008; Springborn et al. 2008] use the so-called logarithmic conformal scale factors. While, the scale on the boundary can be prescribed, these approaches allow only for a conformal change of the metric, which greatly limits the space of plausible maps, and typically leads to excessive variation in scale.

[Myles and Zorin 2012; Myles and Zorin 2013] utilize conformal metric flattening [Ben-Chen et al. 2008] for automatic placement of cone singularities in order to reduce isometric distortion. The ARAP [Liu et al. 2008] energy is approximated based on the conformal factor. We propose to work with the discrete metric directly (edge lengths squared), with access to the complete space of valid metrics, where smaller portions of the space (metrics with low distortion) can be picked easily by posing convex constraints.

Closely related methods to ours are [Chen et al. 2013] and [Chien et al. 2016] which utilize the metric tensor for shape interpolation of simply-connected planar domains. The main idea in [Chen et al. 2013] is that linearly blending the metric tensor of two flat surfaces produces a new metric (a symmetric matrix) whose condition number is bounded by that of the input. The condition number of the metric is a measure for conformal distortion. Chen et al. observed that while in general, the blended metric will not be flat, it can be flattened conformally, to save the conformal distortion bounds. [Chien et al. 2016] on the other hand, deals with C^∞ harmonic planar maps. They are able to maintain isometric distortion bounds, in addition to conformal distortion bounds. Their interpolation scheme blends the smooth metric only on the boundary of the domain. Nonetheless, for harmonic maps, that is sufficient to ensure that the distortion is globally bounded [Chen and Weber 2015].

3 Mathematical Background

In the interest of being somewhat self-contained, we review in this section some basic facts on geometric distortion measures, their relation to the pullback metric tensor, and on convexity/quasiconvexity. For more detailed information on these topics, we refer to [Chien et al. 2016], [Lee 1997], and [Boyd and Vandenberghe 2004], respectively.

3.1 Geometric Distortion

Consider a linear map $T : \mathbb{R}^2 \to \mathbb{R}^2$ given by a matrix A, so that $T(\mathbf{v}) = A\mathbf{v}$ for $\mathbf{v} \in \mathbb{R}^2$. T maps the unit circle $\mathbb{S}^1$ to an ellipse $T(\mathbb{S}^1)$ centered at the origin. The lengths of the axes and the eccentricity of this ellipse reflect the geometric distortion of T. The lengths are given by the square roots of the eigenvalues of $AA^\top$ and $A^\top A$, which share the same spectrum. These values are the *singular values* of A and are denoted σ_a and σ_b, with $\sigma_a \geq \sigma_b$. Clearly, σ_a is the length of the major axis, while σ_b is that of the minor axis. See Figure 2 for illustration. The two singular values together quantify the *isometric distortion*, while their ratio, denoted $K = \frac{\sigma_a}{\sigma_b} \in [1, \infty)$, quantifies the eccentricity of $T(\mathbb{S}^1)$ and the *conformal distortion*. Another common quantity to measure conformal distortion is the small dilatation $k \in [0, 1)$ which is related to K monotonically by $k = (K-1)/(K+1)$ and is 0 iff the map is conformal.

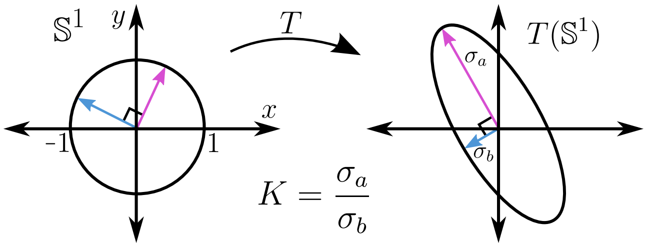


Figure 2: *Local distortion. The unit disk is distorted to an ellipse under a linear map. The lengths of the major and minor axes are the singular values σ_a and σ_b respectively.*

3.1.1 Local Distortion for Differentiable Maps

More generally, we may consider a C^1 map $f : \Omega \subseteq \mathbb{R}^2 \to \mathbb{R}^2$ and consider these distortion quantities for the Jacobians J_f at each point in the domain Ω. These offer a pointwise description of the local geometric distortion of f.

The singular values of the Jacobian at a point are easily expressed in terms of the Wirtinger derivatives, $f_z = \frac{1}{2}(f_x - \mathrm{i}f_y)$ and $f_{\bar{z}} = \frac{1}{2}(f_x + \mathrm{i}f_y)$:

$$\sigma_a = |f_z| + |f_{\bar{z}}|, \quad \sigma_b = \Big| |f_z| - |f_{\bar{z}}| \Big|.$$

For orientation-preserving locally-injective maps, we have that $\det(J_f) = |f_z|^2 - |f_{\bar{z}}|^2 \geq 0$, so the second equation reduces to $\sigma_b = |f_z| - |f_{\bar{z}}|$. For a review of these facts and Wirtinger derivatives and their geometric meaning, we refer to Section 3 of [Chien et al. 2016].

3.2 The Metric Tensor and Geometric Distortion

A metric tensor on a region Ω gives a local notion of distance, and a differentiable planar map f, as above, may be used to assign such a metric tensor via the pullback construction of differential topology. Intuitively, this gives the domain Ω the geometry of $f(\Omega)$ under the standard Euclidean metric tensor. For example, if Ω is a triangle and f is an affine map, then Ω with the pullback metric tensor will have the geometry of the image triangle $f(\Omega)$, i.e., the same edge lengths and angles. This example will be quite relevant for our basic discrete setting, described further in Section 4.1.

More concretely, the pullback metric tensor is denoted by M_f and is easily calculated in terms of the Jacobian: $M_f := J_f^\top J_f$. For brevity, we refer to the pullback metric tensor as simply the *metric tensor*, as the map under consideration is usually implicit. Given that the singular values of a matrix A are determined by the eigenvalues of $A^\top A$, it is not surprising that the distortion measures for f are nicely expressed in the entries of this metric tensor.

3.2.1 Distortion In Terms of Metric Tensor Entries

Instead of using the actual entries of M_f, which are not invariant to precomposition with isometries of $\mathbb{R}^2$ (isometric choices of coordinates or charts on $\mathbb{R}^2$), let us first develop some more meaningful coordinates that are at least partially invariant to these choices. Using the additive decomposition of J_f into similarity and anti-similarity parts [Chien et al. 2016], we obtain a formula in terms of f_z, $f_{\bar{z}}$, and a third quantity $\eta := f_{\bar{z}}\overline{f_z}$:

$$M_f = \begin{pmatrix} |f_z|^2 + |f_{\bar{z}}|^2 & 0 \\ 0 & |f_z|^2 + |f_{\bar{z}}|^2 \end{pmatrix} + 2\begin{pmatrix} \mathrm{Re}\,(\eta) & \mathrm{Im}\,(\eta) \\ \mathrm{Im}\,(\eta) & -\mathrm{Re}\,(\eta) \end{pmatrix}.$$

In light of this expression, we define the useful quantity: $\mathcal{A} := |f_z|^2 + |f_{\bar{z}}|^2$, which is invariant under choice of charts. Note that $\mathcal{A} = \|J_f\|_F^2/2$ where $\|\cdot\|_F$ denotes the Frobenius norm. We also note that η is a geometrically relevant quantity which has the same complex argument as the Beltrami coefficient $\mu = f_{\bar{z}}/f_z$, which encodes the (local) maximal stretch direction. Furthermore, $|\eta|$ is invariant under choice of charts, and as $|\eta| \to 0$ the map becomes more and more conformal, locally. See [Chien et al. 2016] for further details on the aforementioned quantities.

The $\mathcal{A}$ and η coordinates are linear in the entries of the metric tensor, and they lead to simple expressions for the isometric and conformal distortion quantities:

$$(\sigma_a)^2 = |f_z|^2 + |f_{\bar{z}}|^2 + 2\,|\eta| = \mathcal{A} + 2\,|\eta| \tag{1}$$

$$(\sigma_b)^2 = |f_z|^2 + |f_{\bar{z}}|^2 - 2\,|\eta| = \mathcal{A} - 2\,|\eta| \tag{2}$$

$$K^2 = \frac{(\sigma_a)^2}{(\sigma_b)^2} = \frac{\mathcal{A} + 2\,|\eta|}{\mathcal{A} - 2\,|\eta|}. \tag{3}$$

We also note here that sometimes the singular values (or their squares) are combined to obtain a single measure that is used to quantify the isometric distortion. Later on we will refer to two of these measures, $\tau := \max(\sigma_a^2, 1/\sigma_b^2)$ and $\beta := \sigma_a^2 + 1/\sigma_b^2$. Bounding σ_a^2 from above and σ_b^2 from below with positive quantities produces bounds from above on these quantities, and vice versa.

The relatively simple form of the above expressions allows us to easily see that they are all convex or quasiconvex in our $\mathcal{A}$ and η coordinates.

3.3 Convexity and Quasiconvexity

The basic facts in this section are presented and simultaneously used to determine the nature of the distortion quantities from the previous section. As the entries of the metric tensor, $\mathcal{A}$ and η, and the edge lengths squared (see Section 4.3) are all linear in terms of each other, convexity/quasiconvexity of the quantities below are the same in all these variables. The mathematical facts are stated without proof or reasoning, but are justified within [Boyd and Vandenberghe 2004].

A convex function $g : R \subseteq \mathbb{R}^n \to \mathbb{R}$ on a convex subset R is a function such that for any pair $p, q \in R$, we have that $g((1-t)p + tq) \leq (1-t)g(p) + tg(q)$ for all $t \in [0, 1]$. The definition of a concave function is the same but with the direction of the inequality reversed.

Linear functions are both convex and concave, so we see that $\mathcal{A}$ above is both convex and concave. Norms on vector spaces are convex, so we see that $|\eta|$ is convex. Multiplying a convex function by a negative scalar results in a concave function, while doing the same to a concave function results in a convex function. Thus $-|\eta|$ is concave. The non-negative weighted sum of convex functions is convex, and such sums of concave functions are concave. With this, we may see from (1) and (2) that σ_a^2 is convex and σ_b^2 is concave.

The reciprocal of a positive concave function is convex. This shows us that $1/\sigma_b^2$ is convex, implying also that β is convex. And finally, the pointwise maximum (or supremum) of a set of convex functions is convex, showing us that τ is convex. Analogously, the pointwise minimum (or infimum) of a set of concave functions is concave, a fact that we'll need later.

A quasiconvex function $g : R \subseteq \mathbb{R}^n \to \mathbb{R}$ on a convex subset R is one such that all its sub-levelsets:

$$S_c = \{x \mid g(x) \leq c\},$$

for all $c \in \mathbb{R}$ are convex. There is an analogous definition for quasiconcave functions. These definitions are generalizations of convexity and concavity, and all convex and concave functions are quasiconvex and quasiconcave, respectively.

The quotient of a positive convex function by a positive concave function is quasiconvex, so we see from (3) that K^2 is quasiconvex. Finally, for a positive quasiconvex/quasiconcave function, any positive power of that function is quasiconvex/quasiconcave. With this we see that σ_a and K are quasiconvex and σ_b is quasiconcave.

4 Bounded Distortion Metrics

In this section, we first describe our basic discrete setting, that of piecewise linear maps on triangular mesh surfaces, and see that distortion will be quantified per triangle. Then, it is seen that the space of bounded distortion metrics for a single triangle is a convex second-order cone space. Finally, we transit to edge length squared coordinates to see that the same holds true for all bounded distortion discrete metrics on a triangular mesh surface. Suitable generalizations to higher dimensions are also mentioned briefly, where appropriate.

4.1 Global Parametrization via Piecewise Linear Maps

Within the background section, we considered the pointwise distortion measures and how they relate to the metric tensor for a differentiable planar map f. Throughout this paper, we work in a discrete setting meant to approximate differentiable maps, that of piecewise linear maps. For such maps from $\mathbb{R}^2 \to \mathbb{R}^2$, as discussed previously, we would take Ω to be triangulated densely and consider maps f that are piecewise linear on the triangles (affine on each triangle individually, and continuous overall). In this setting, the Jacobians, and thus the metric tensors and distortion quantities, are constant across each individual triangle. Within the rest of this work, these quantities and other related ones are specified per triangle, instead of pointwise over the domain to be parametrized.

For our work, the domain to be parametrized will not be planar; it will be a triangular mesh surface S in $\mathbb{R}^3$. Below, we review some basic definitions and constructions relevant to this scenario and the discussion is modelled partially on Section 3 in [Myles and Zorin 2012]. Further details are contained there, and in its cited references [Springborn et al. 2008; Ben-Chen et al. 2008; Lai et al. 2010].

Let g and m denote the genus and number of boundary components of S. Unless $g = 0$ and $m > 0$, topological considerations imply that there is no continuous locally injective map of S to $\mathbb{R}^2$. Thus, our goal is to obtain a *global parametrization* of S, which is a selection of cuts transforming S into a mesh disc S_c, and a piecewise linear parametrization map $f : S_c \to \mathbb{R}^2$. We use Δ_i to refer to the triangles of both S and S_c, between which there is an obvious correspondence.

The parametrization map f induces a metric tensor on each triangle Δ_i given by the pullback of the standard Euclidean metric by $f_i := f|_{\Delta_i}$. Any edge within the cuts has two images in S_c and for an arbitrary parametrization map f, there is no guarantee that their lengths will agree in the induced metrics. If f does result in a metric which agrees on all these pairs of edges, then it determines a *discrete metric* on S. This is simply a cone metric on S whose cone points are a subset of the vertices of S.

Conversely, if we consider a discrete metric on S and make a selection of cuts that includes all the cone points, then the resulting discrete metric on S_c has all pairs of images of cut edges having length equality. This metric is induced by a parametrization map f that is unique up to post-composition by isometries of $\mathbb{R}^2$. As the cuts include all the cone points, the metric is flat at internal vertices of S_c and f is locally injective. Finally, note the important fact that the initial choice of discrete metric on S fully determines the distortion of Δ_i by f, and the later choice of cuts does not affect it at all.

Thus, we work in the realm of discrete metrics on S and restrict to metrics of bounded distortion. The final parametrization is then obtained by laying the triangles one-by-one in the parametric domain by traversing the dual graph of the mesh [Springborn et al. 2008]. The result is automatically locally injective by flatness at internal vertices.

For the coming discussions and implementation of our method, we need explicit realizations of the metric tensors on each triangle. For this, we may consider each Δ_i and its eventual image $f(\Delta_i)$, individually. We make choices of local orthonormal coordinates on each to obtain isometric charts g_i and h_i. It is key to note that post-composition and pre-composition by rigid isometries leave the distortion quantities invariant, so the choice of these charts is somewhat arbitrary. For derivation of relevant formulae, we use charts such that $h_i \circ f|_{\Delta_i} \circ g_i^{-1} : g_i(\Delta_i) \subset \mathbb{R}^2 \to \mathbb{R}^2$ is the linear map illustrated in Figure 3. For the rest of the work, we use J_f^i, M_f^i, σ_a^i, σ_b^i, and K_i to denote the Jacobians, metric tensors, and distortion quantities, respectively, of the composition of maps above. Superscript and subscript i's will also be appended to other relevant quantities associated with the map above.

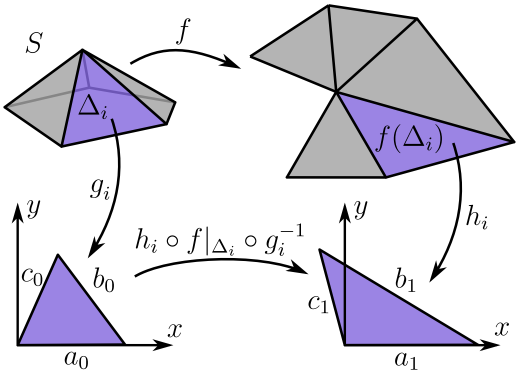


Figure 3: *Charts illustration.*

4.2 The Bounded Distortion Space for Δ_i

With all our relevant quantities being determined per triangle, let us consider the space of bounded distortion metrics for a single triangle. With the $\mathcal{A}_i$ and η_i coordinates developed in Section 3.2.1, we get a particularly nice picture of this space.

Note first from the defining equations that $\mathcal{A}_i \geq 0$ and $\eta_i \in \mathbb{C}$. In addition, the metric tensor has to be positive-definite, which corresponds to the condition: $\mathcal{A}_i > 2|\eta_i|$. Thus the space of all valid metrics is a simple convex cone.

Now let us see what subset of this space is selected by the distortion inequalities. If we'd like a bound on σ_a^i, then this is equivalent to asking that $(\sigma_a^i)^2 = \mathcal{A}_i + 2\,|\eta_i| < C_a$ for some $C_a > 0$, which is equivalent to $\mathcal{A}_i < C_a - 2\,|\eta_i|$. Similarly, for a bound on σ_b^i, we have $\mathcal{A}_i > C_b + 2\,|\eta_i|$ for some $C_b > 0$. Note that positive-definiteness is guaranteed by the condition of having σ_b bounded from below.

For conformal distortion, we have that this is equivalent to asking for $K_i^2 = \frac{\mathcal{A}_i + 2|\eta_i|}{\mathcal{A}_i - 2|\eta_i|} < C_K$ for some $C_K \in (1, \infty)$. Simple calculation gives:

$$\mathcal{A}_i > 2\,|\eta_i| \left(\frac{C_K + 1}{C_K - 1} \right).$$

Note that positive-definiteness is also guaranteed by this condition.

Viewing things in the $\mathcal{A}_i$ and $|\eta_i|$ plane (see Figure 4), we see that the space of bounded distortion metrics for a single triangle is a convex second-order cone space.

4.2.1 Generalization to Higher Dimensions

We have seen that the space of bounded distortion metrics for a single triangle is a second-order cone space in the entries of the metric tensor. We were guided by the intuition that the singular values of A are obtained from the eigenvalues of $A^\top A$. We note briefly here that the same intuition is of value in higher dimensions, when we are considering the space of bounded distortion metrics for a single n-simplex in $\mathbb{R}^n$ for any $n \geq 2$.

More specifically, suppose we have a linear map given by a nonsingular matrix A that is distorting our n-simplex. The isometric

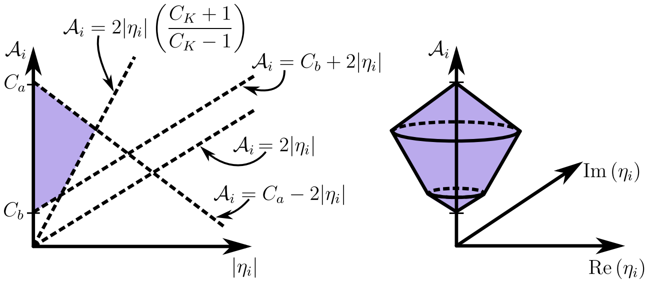

$\mathcal{A}_i = 2|\eta_i|\left(\frac{C_K+1}{C_K-1}\right)$
$\mathcal{A}_i$
C_a
$\mathcal{A}_i = C_b + 2|\eta_i|$
$\mathcal{A}_i = 2|\eta_i|$
C_b
$\mathcal{A}_i = C_a - 2|\eta_i|$
$|\eta_i|$
$\mathcal{A}_i$
$\mathrm{Im}\,(\eta_i)$
$\mathrm{Re}\,(\eta_i)$

Figure 4: *The bounded distortion space for Δ_i. Left: a radial cross section visualized as intersection of three half-planes (purple), one from each of our distortion bounds. Right: the full space as a volume of revolution around the $\mathcal{A}_i$ axis.*

distortion is quantified by the largest and smallest of the n eigenvalues of the metric tensor $A^\top A$ and the ratio of these eigenvalues quantifies the conformal distortion. Recall these eigenvalues are squares of the singular values of A. By bounding the eigenvalues from above and below, and by bounding their ratio from above, we obtain a space of bounded distortion metrics for our n-simplex.

Theorem 1. *The space of bounded distortion metrics for an n-simplex, $n \geq 2$, is convex in the entries of the metric tensor.*

Proof. The largest and smallest eigenvalues of $A^\top A$, denoted $\lambda_{\max}$ and $\lambda_{\min}$ satisfy the following equations:

$$\lambda_{\max} = \max_{\mathbf{u}\in\mathbb{S}^{n-1}} \langle \mathbf{u}, A^\top A\mathbf{u}\rangle$$
$$\lambda_{\min} = \min_{\mathbf{u}\in\mathbb{S}^{n-1}} \langle \mathbf{u}, A^\top A\mathbf{u}\rangle,$$

where $\langle \mathbf{u}, A^\top A\mathbf{u}\rangle$ is linear in the entries of $A^\top A$, so is both convex and concave. Thus $\lambda_{\max}$ is convex as a maximum of convex functions, and $\lambda_{\min}$ is concave as the minimum of concave functions. As $A^\top A$ is positive-definite, both of these values are positive, and their ratio is quasiconvex. □

The theorem has been noted before (without proof) outside the context of bounded distortion maps [Boyd and Vandenberghe 2004, pg. 407]. The brief proof we have shown is a folklore proof that is well-known to many. However, noting the connection to computation of a parametrization is completely novel. More about utilizing this information is in Section 10.1. Finally, we note that convexity by itself does not immediately imply that the problem can be solved efficiently. However, as seen in Section 4.2, for a 2-simplex (a triangle), the convexity is characterized by second-order cones for which efficient solvers are available [ApS 2015].

4.3 Edge Length Squared Coordinates

To obtain a discrete metric on S, we also need that the metric tensors on each individual triangle agree on shared edges. It is not too hard to see that if Δ_i and Δ_j denote two triangles that share an edge, the length equality under the different metric tensors M_f^i and M_f^j is linear in the entries of these metric tensors. As noted in [Chen et al. 2013], the edge length squared of a triangle Δ_i is given by $\mathbf{v}^\top M_f^i \mathbf{v}$ ($\mathbf{v}$ is the edge vector), which is a linear expression in the entries of M_f^i.

If we were to try to use the metric tensor entries, or $\mathcal{A}_i$ and η_i, explicitly to describe the space of discrete metrics on S, then we would need to add additional linear equalities to ensure this length consistency. To avoid this, and to make these equalities implicit, we use the edge lengths squared as variables.

Below, we present the expressions for $\mathcal{A}_i$ and $|\eta_i|^2$ in terms of the edge lengths squared, as they exhibit a nice symmetry that is reflective of the fact that they are independent of the choice of charts. Both these and the expressions for $\mathrm{Re}\,(\eta_i)$ and $\mathrm{Im}\,(\eta_i)$ are derived with the choice of charts mentioned previously and some basic trigonometry. As $\mathrm{Re}\,(\eta_i)$ and $\mathrm{Im}\,(\eta_i)$ are not independent of the choice of charts, they lack this symmetry and are less enlightening, so are contained within Appendix A.

Let a_0, b_0, c_0 denote the edge lengths *squared* of Δ_i, and let a_1, b_1, c_1 denote the lengths squared of the corresponding edges of $f(\Delta_i)$ (the induced edge lengths squared of Δ_i). We choose to use this notation, without the superscript 2's, as it makes linearity of the expressions for $\mathcal{A}_i$ and η_i more apparent.

Before giving the formulae, we also define a useful quantity for compactly expressing these values, $\mathscr{T} := \sqrt{2a_0b_0 + 2a_0c_0 + 2b_0c_0 - a_0^2 - b_0^2 - c_0^2}$, and note that it is four times the area of Δ_i in S.

$$\mathcal{A}_i = \frac{1}{\mathscr{T}^2}\Big(\left(-a_0 + b_0 + c_0\right) a_1 + \left(a_0 - b_0 + c_0\right) b_1$$
$$+ \left(a_0 + b_0 - c_0\right) c_1 \Big)$$
$$|\eta_i|^2 = \frac{1}{\mathscr{T}^4}\Big(\left(b_0c_0\right) a_1^2 + \left(a_0c_0\right) b_1^2 + \left(a_0b_0\right) c_1^2$$
$$+ \left(c_0(-a_0 - b_0 + c_0)\right) a_1b_1$$
$$+ \left(b_0(-a_0 + b_0 - c_0)\right) a_1c_1$$
$$+ \left(a_0(a_0 - b_0 - c_0)\right) b_1c_1 \Big).$$

For both expressions above, if one permutes the lengths squared a_j, b_j, c_j, we see that the expressions remain unchanged. In addition to this symmetry, note the linearity of $\mathcal{A}_i$ in a_1, b_1, c_1; a property that is shared by the expressions for $\mathrm{Re}\,(\eta_i)$ and $\mathrm{Im}\,(\eta_i)$ in the appendix. This is expected as it is the inverse to the linear expressions for the edge lengths squared in terms of the metric tensor entries (which are linear in $\mathcal{A}_i$ and η_i).

Before moving on, let us note that these expressions allow us to impose bounded distortion conditions in the edge length squared coordinates. Furthermore, if three edge lengths squared of a triangle Δ_i satisfy some bounded distortion conditions, then the edge lengths are guaranteed to satisfy the triangle inequalities.

To see this, note that we have non-singular linear maps going back and forth from the $\mathcal{A}_i$ and η_i coordinates to the edge length squared coordinates. If we have a positive-definite metric, with $\mathcal{A}_i > 2|\eta_i|$, then the induced edge lengths squared will satisfy the triangle inequalities (or their square roots will, rather). Furthermore, taking any triple of edge lengths satisfying the triangle inequalities, we may achieve its edge lengths squared by integrating some positive-definite metric tensor. Thus, we see that these linear maps create a bijection between the second-order cones given by $\mathcal{A}_i > 2|\eta_i|$ and the triangle inequalities (remember, they are not linear inequalities in the edge length squared coordinates). If bounds are chosen on the distortion quantities, we get subsets of these cones that are in bijection with each other. Thus these bounds imply satisfaction of the triangle inequalities.

4.3.1 The Bounded Distortion Space for S

With the expressions above, we have a full description of the bounded distortion space of discrete metrics on S. From the initial viewpoint of $\mathcal{A}_i$ and η_i coordinates for each triangle, the space would be the product of many cone spaces, one for each triangle,

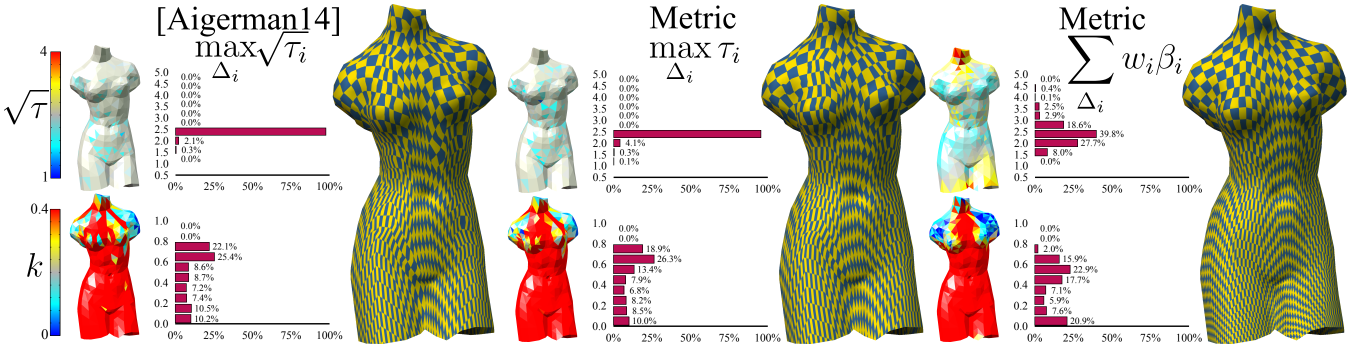


Figure 5: *Symmetric isometric energies. A comparison of variants V1 and V2 of our method with [Aigerman et al. 2014], all optimizing symmetric isometric energies. The energies optimized in the two leftmost examples are equivalent (see Section 6.2.2). The mesh is a topological disk, and lacks cones and cuts to stress test the methods. The left two results are nearly identical and serve as evidence of optimality (for both methods). The result on the right shows our metric optimization for the β measure (Sections 6.2.1 and 8.1). The maximum isometric distortion is higher but the average distortion is lower and the map is smoother.*

with additional linear length (squared) equalities imposed, one for every interior edge. The result is a convex second-order cone space of dimension $3|F| - |E_{\text{int}}| = |E|$, where F denotes the set of triangles of S, and E and E_{int} denote the sets of edges and interior edges, respectively.

With the use of edge length squared coordinates, we get a linear reparametrization of this space, and the dimensionality of the space is clearer. Each bounded distortion inequality is still a second-order cone inequality, so it's also clear in this setting that we have a convex second-order cone space.

Finally, we note briefly here that Theorem 1 is easily used in the same fashion to generalize this convexity to higher dimensions. Given an n-dimensional simplicial complex S_n with manifold carrier and an initial discrete metric M_0, the space of bounded distortion discrete metrics may be seen as the product of the convex bounded distortion spaces for each n-simplex, intersected with additional linear length squared equalities. There is also an analogous parametrization of this space with edge length squared coordinates.

Theorem 2. *The space of bounded distortion discrete metrics for (S_n, M_0) is convex in the entries of the metric tensors on each n-simplex (and thus in the edge lengths squared as well).*

5 Curvature Constraints

In this brief section, we describe the final set of constraints on our desired space of discrete metrics, the curvature constraints. We want to be able to prescribe particular vertices of S as cone points and the cone angles at these vertices. Without such prescriptions, an arbitrary bounded distortion discrete metric will likely have every vertex as a cone point. This would require an extremely complicated cut to be performed before obtaining the global parametrization map. So, we enforce a smaller selection of cone points and cone angles. The values will need to satisfy the discrete Gauss-Bonnet equation.

To be precise, we are using the angle deficit notion of discrete Gaussian curvature, which resides at the vertices of a cone metric. Let $\mathfrak{K}(v)$ and α_v denote the discrete curvature and angle sum, respectively, at v.

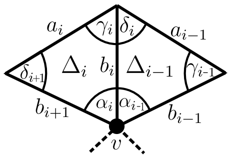
$$\mathfrak{K}(v) = \begin{cases} 2\pi - \alpha_v, & \text{if } v \text{ is an interior vertex} \\ \pi - \alpha_v, & \text{if } v \text{ is a boundary vertex} \end{cases}$$

A cone metric on a topological surface $\Sigma_{g,m}$ (with genus g and m boundary components) will obey the discrete Gauss-Bonnet equation. Let V denote the set of vertices (or cone points), and χ denotes the Euler characteristic.

$$\sum_{v \in V} \mathfrak{K}(v) = 2\pi\chi(\Sigma_{g,m}) = 2\pi(2 - 2g - m).$$

When we set the desired cone points and curvatures at the cone points, they must satisfy the above equation.

We finish the section with the explicit expressions for the curvature at a vertex v in terms of the edge lengths squared of the discrete metric. To set notation, consider the depicted diagram. The α_i denote the angles that make up the angle sum at v and the a_i and b_i denote the edge lengths *squared* for triangles having v as a vertex. Then the law of cosines gives us the following formula:



$$\mathfrak{K}(v) = 2\pi - \sum_i \alpha_i, \text{ where} \tag{4}$$

$$\alpha_i = \arccos\left(\frac{b_i + b_{i+1} - a_i}{2\sqrt{b_i b_{i+1}}}\right).$$

The expression for boundary cone points is the same, except 2π is replaced with π. These expressions are clearly nonlinear and we deal with them with sequential convex optimization (Section 7).

6 The Constrained Optimization Problem

The curvature constraints complete our framework, formulating a constrained optimization problem that will help us achieve bounded distortion global parametrization. We first state this general problem as the basic setup of our framework, and then describe its great flexibility by discussing many of its variants.

6.1 The Framework

To state the framework precisely, we reiterate and set some new notation. Let V, E, F denote the sets of vertices, edges, and faces (triangles) in S. Furthermore, let $\mathbf{e} = (d_1, d_2, \ldots, d_{|E|})$ denote the edge lengths *squared* of the edges $e_1, e_2, \ldots, e_{|E|}$. In addition, we let $\sigma_a^i, \sigma_b^i, K_i$ denote the respective distortion quantities for triangular face Δ_i, and let C_a, C_b, C_K denote the desired global bounds

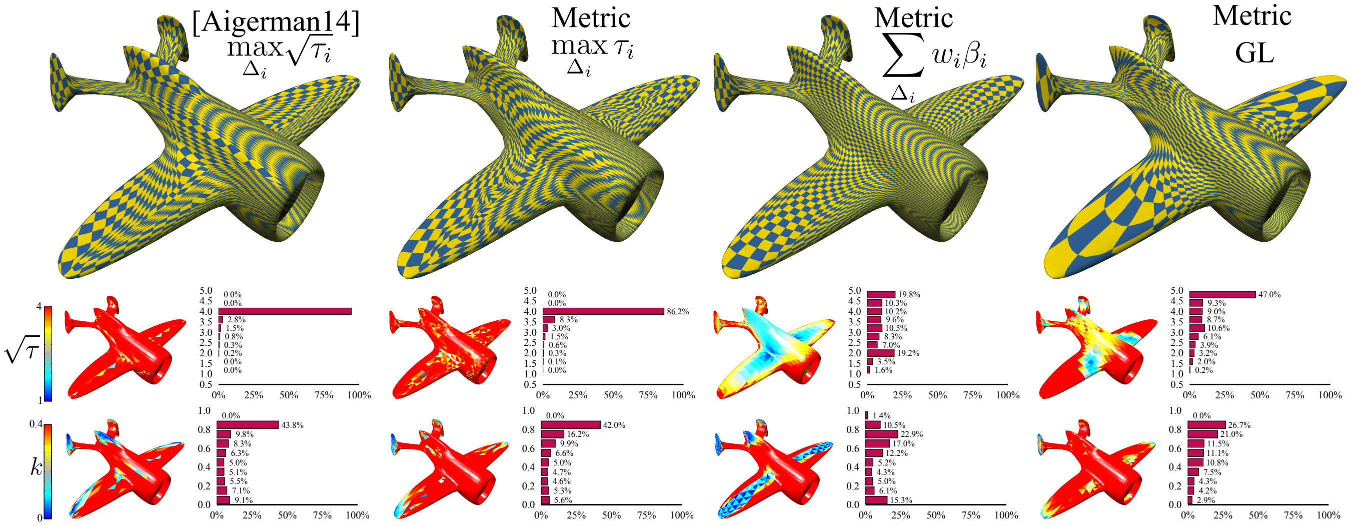


Figure 6: *Robust isometric energies. The aircraft model is a disk with no cones. The single boundary is the circular part at the front and mapping it bijectively to the plane is highly challenging due to the extreme protrusions of the wings and tail. We compare the results of our metric optimization using three types of isometric energies. The two images on the left show minimization of the maximal isometric distortion which were produced with Aigerman's method and variant V1 of our method (the energies optimized are equivalent; see Section 6.2.2). The next two are the results of variants V2 and V3, optimizing β and the Green Lagrange energy, respectively (Sections 6.2.1 and 8.1). For reference, we also ran [Springborn et al. 2008] which failed to converge and LSCM which produces a map with τ as high as 6000.*

for these quantities. Lastly, we let $\mathfrak{K}_j$ and $\mathfrak{K}_j^0$ denote the discrete curvature of the cone metric and the desired discrete curvature at vertex v_j, respectively; and let $V_c \subseteq V$ denote the points of S where we'd like to have curvature constraints.

$$\begin{aligned}
&\underset{\mathbf{e}\in\mathbb{R}^{|E|}}{\text{minimize}} && E(\mathbf{e}) && && (5)\\
&\text{subject to} && (\sigma_a^i)^2(\mathbf{e}) \leq C_a, && i = 1,\ldots,|F| && (6)\\
& && (\sigma_b^i)^2(\mathbf{e}) \geq C_b, && i = 1,\ldots,|F| && (7)\\
& && K_i^2(\mathbf{e}) \leq C_K, && i = 1,\ldots,|F| && (8)\\
& && \mathfrak{K}_j(\mathbf{e}) = \mathfrak{K}_j^0, && j = 1,\ldots,|V_c| && (9)
\end{aligned}$$

Above, we have purposefully left E, the objective function, as unspecified, as we have many potential choices for it. This will be discussed in the next section.

6.2 Framework Flexibility

The standard setup above is quite flexible and accommodates many different formulations to attack different problems. The strength of our variables is reflected in the many quantities that are nicely expressed with them. Most of the quantities listed below are highly nonlinear in spatial coordinates, while they are convex or quasiconvex in our coordinates.

Given our setting of piecewise linear maps, we discuss these quantities first per triangle Δ_i, adding a subscript or superscript i to emphasize this. To obtain the quantities over the entirety of S as a function of the edge lengths squared, we may either take a weighted sum with relative areas of the triangles as the weights, or we may take a maximum over all the triangles of S. Examples may be seen in Section 8.

6.2.1 Convex Quantities

Before listing several quantities that are linear, quadratic, or more generally convex in the entries of the metric tensor (and thus the edge lengths squared), we note that the summed quantities and the maximums (over S) will also be convex as a result. These may then be used as either energies to be optimized, or as additional convex bounds in the framework, giving us many variations of the basic framework. For each quantity, we also note whether they will tend to produce near-isometric or near-conformal maps when minimized or bounded. Such quantities will be referred to as isometric or conformal measures, respectively.

First, we note the Green-Lagrange tensor known in mechanics, introduced to the graphics community for texture mappings in [Maillot et al. 1993]:

$$E_{GL}^i = \|M_f^i - I\|_F^2.$$

The quantity is an isometric measure, as E_{GL}^i is locally zero when the metric tensor equals the identity matrix I (implying that the Jacobian is a rotation and the map is an isometry). It is nonlinear and nonconvex when expressed in the spatial coordinates [Botsch et al. 2010, Section 5.5.2], with many local minima, but fortunately, it is a convex quadratic function in our variables. To see this, express E_{GL}^i in terms of the singular values and substitute with the expressions given in Equations (1) and (2):

$$\begin{aligned}
E_{GL}^i &= \left((\sigma_a^i)^2 - 1\right)^2 + \left((\sigma_b^i)^2 - 1\right)^2\\
&= (\mathcal{A}_i + 2|\eta_i| - 1)^2 + (\mathcal{A}_i - 2|\eta_i| - 1)^2\\
&= 2\left((\mathcal{A}_i - 1)^2 + 4|\eta_i|^2\right).
\end{aligned}$$

Another example of a popular isometric measure is β which was defined in Section 3.2.1:

$$\beta_i = (\sigma_a^i)^2 + \frac{1}{(\sigma_b^i)^2}.$$

It is nonconvex in the spatial coordinates. [Aigerman et al. 2014] considered the square root of this measure, and convexified for each iteration of their optimization. A closely related quantity was also optimized in [Schreiner et al. 2004] by successive independent vertex relaxations with random directions and a line search. As noted in Section 3.3, β_i is convex in our variables.

We also highlight the important fact that this measure is *symmetric*, where symmetry of a distortion quantity E_i is defined by having the same value for f_i and f_i^{-1}, which simply inverts the singular values and swaps their order, i.e. $E_i(\sigma_a, \sigma_b) = E_i(\frac{1}{\sigma_b}, \frac{1}{\sigma_a})$. This ensures that the measure does not favor shrinkage over expansion and it penalizes drastically collapsed elements, since the energy explodes when the triangle area vanishes.

The isometric measure τ_i is another symmetric measure which is convex in our coordinates, also noted in Section 3.3:

$$\tau_i = \max\left((\sigma_a^i)^2, \frac{1}{(\sigma_b^i)^2}\right).$$

The square root of this quantity was the quantity considered in [Sorkine et al. 2002] which used it to measure the isometric distortion, but did not optimize for it directly. The same energy was also accommodated by [Aigerman et al. 2014], and as their approach dictates, had to be convexified at each iteration. We may optimize for τ_i directly in our coordinates (or bound it and add it to our convex bounds).

We note one more quantity, which is obviously convex:

$$E_{CONF}^i = \frac{1}{2}\left((\sigma_a^i)^2 - (\sigma_b^i)^2\right) = 2|\eta|.$$

This is quite similar to LSCM [Lévy et al. 2002], and is a conformal measure (Figure 7). Finally, we note that squaring E_{CONF}^i is a convex quadratic function in our variables, further simplifying it.

6.2.2 Quasiconvex Quantities

For quasiconvex per triangle quantities, we recall that the maximum over the triangles of S will be quasiconvex, but the weighted sum, in general, will not be. So for these quantities, we may utilize only the maximum over triangles as energies to optimize or as bounds to add. Quasiconvex programming [Eppstein 2005] is needed to optimize such energies, which is computationally less efficient than convex programming but is still tractable. Moreover, any of these quasiconvex quantities may still effectively be utilized as a bound in our framework, as these bounds are in fact convex.

We note first that the quantities $\sqrt{\beta_i}$ and $\sqrt{\tau_i}$ considered in [Aigerman et al. 2014] and [Sorkine et al. 2002] are quasiconvex, so if we desire exact replication of these, we may accomplish this, albeit with the penalties mentioned above. Furthermore, we note the following equalities:

$$\max_{\Delta_i}\sqrt{\beta_i} = \sqrt{\max_{\Delta_i}\beta_i} \quad \& \quad \max_{\Delta_i}\sqrt{\tau_i} = \sqrt{\max_{\Delta_i}\tau_i}.$$

The practical meaning of these equalities is that since the square root is a monotonic function, optimizing for the convex energies (without the square roots) or the quasiconvex ones is equivalent (as they have the same minimizers).

Now, let us note briefly two popular quantities from the literature that are quasiconvex in our coordinates. The first is the ARAP [Liu et al. 2008] energy, an isometric measure:

$$E_{ARAP}^i = \min_{R\in SO(2)} \frac{1}{2}\|J_f^i - R\|_F^2.$$

This energy was also considered in [Levi and Zorin 2014], where the Frobenius norm was substituted with the L_∞ norm, i.e., the maximum over the triangles was optimized for (so-called $ARAP-L_\infty$ energy).

The second quantity is the LSCM energy [Lévy et al. 2002; Liu et al. 2008], a conformal measure:

$$E_{LSCM}^i = \frac{1}{2}(\sigma_a^i - \sigma_b^i)^2.$$

The ARAP energy is nonconvex in the spatial variables, while the LSCM energy is quadratic and convex there (one of the few popular distortion measures that takes a simpler form when expressed in spatial coordinates). For the sake of brevity, a proof sketch for quasiconvexity of both quantities is relegated to Appendix B.

Finally, while we already use K_i^2 as a bound in our standard setup, we note that $\max\limits_{\Delta_i} K_i^2$ and $\max\limits_{\Delta_i} K_i$ are quasiconvex conformal energies. These are perhaps better measures of conformal distortion than $|\eta_i|$, as they are symmetric and K_i is used as the de facto definition of conformal distortion for smooth conformal maps. With minimization of these, the method should produce *extremal* quasiconformal maps. However, we assert that due to sensitivity to outliers, in the discrete case, a better approach for extremal quasiconformality should be based on Teichmüller maps [Weber et al. 2012] or strict minimizers [Levi and Zorin 2014].

6.2.3 Additional Variants

In addition to utilizing the sums and maximums over S of the above quantities as energies and convex bounds, there are several other variants of the framework that may be made to address different scenarios.

We note first that with several of the quantities mentioned in the previous two sections, if we are to minimize their maximum over the triangles of S, then we may reasonably drop bounds from the standard setup. For example, if $\max_{\Delta_i}(\tau_i)$ is to be minimized, we would drop the bounds on the isometric distortion constants σ_a^i and σ_b^i since the tighter bound for these will be obtained automatically. More details are in Section 8.

We may also project a given reference map to a bounded distortion parametrization if the reference map is not satisfactory [Aigerman and Lipman 2013; Kovalsky et al. 2015]. Let $\mathcal{A}_0^i$ and η_0^i denote these various values for the reference map, and consider the following quadratic energy:

$$E_{GLD}^i = (\mathcal{A}_i - \mathcal{A}_0^i)^2 + 4|\eta_i - \eta_0^i|^2, \tag{10}$$

which strives to make the metric tensors of the reference map and the projected one, as close as possible (in the least squares sense). For the choice of identity map as the reference map, we get $E_{GL}^i = E_{GLD}^i$. Note that due to the second term, it is essential to utilize the same choice of parametrization chart to determine the η_i and η_0^i values. Furthermore, due to convexity, other norms such as L_1 or L_∞ can be used instead of the L_2 norm above.

Lastly, we note that if different bounds on the distortion quantities (or any of the convex or quasiconvex quantities mentioned) are desired in different regions of S, then they may be specified per triangle. To accomplish this, we could generalize the standard setup, replacing C_a, C_b, C_K in bounds (6) - (8), letting the trianglewise bounds be denoted by C_a^i, C_b^i, C_K^i, and replacing C_a, C_b, C_K with these in bounds (6) - (8), respectively.

7 Sequential Convex Programming

Once the variant of our constrained optimization problem is chosen, we attempt to solve it with sequential convex programming

(SCP). We give a brief overview of this useful method and describe how it specializes to our framework. For details and more complete expositions on SCP, we refer to [Dinh and Diehl 2010].

SCP aims to solve nonconvex problems of the sort:

$$\begin{aligned} &\underset{x\in\mathbb{R}^n}{\text{minimize}} && E(x) && \\ &\text{subject to} && \phi_i(x) \le 0, && i = 1, \dots, p \\ & && \psi_i(x) = 0, && i = 1, \dots, q \end{aligned}$$

where all these functions are real-valued, E and the ϕ_i may be nonconvex, and the ψ_i may be nonlinear. In most cases, ψ_i need to be sufficiently differentiable as well.

The basic approach to doing this is to begin at a reasonable guess x_0, and then to iteratively solve a convex approximation to this problem to obtain a sequence x_j. As support for this approach, we note that under certain optimality conditions, convergence to a local minimum is guaranteed with a good initial guess [Dinh and Diehl 2010].

We now outline this approach more specifically. Let $\hat{E}_j$ denote a convex approximation to E near x_j, let $\hat{\phi}_i^j$ denote the same for ϕ_i, and let $\hat{\psi}_i^j$ denote a linear approximation to ψ_i near x_j. Then the convex subproblem we solve to obtain x_{j+1} is the following:

$$\begin{aligned} &\underset{x\in\mathbb{R}^n}{\text{minimize}} && \hat{E}_j(x) && \\ &\text{subject to} && \hat{\phi}_i^j(x) \le 0, && i = 1, \dots, p \\ & && \hat{\psi}_i^j(x) = 0, && i = 1, \dots, q. \end{aligned}$$

The realm of application for SCP is quite broad, but our problem and its variants are considerably simpler. In particular, for us, the function to be optimized, E, is already convex, as are the inequality constraints (the ϕ_i), inequalities (6) - (8). Thus, to set up each subproblem, we only require linear approximations of the equality constraints (the ψ_i), equations (9). The subproblems are second-order cone programs which can be solved efficiently using primal-dual interior point method [ApS 2015].

7.1 Linearization of the Curvature Constraints

To formulate the convex subproblem, we need to linearize the nonlinear curvature constraints at $\mathbf{e}_l$ (we use l to index our sequence of points to avoid conflict with the index in (9)). First, let us consider the affine approximation:

$$\vec{\mathfrak{K}}(\mathbf{e}) := \begin{pmatrix} \mathfrak{K}_1(\mathbf{e}) \\ \vdots \\ \mathfrak{K}_{|V_c|}(\mathbf{e}) \end{pmatrix} \approx \hat{\mathfrak{K}}^l(\mathbf{e}) := \vec{\mathfrak{K}}(\mathbf{e}_l) + J_{\mathfrak{K}}^l(\mathbf{e} - \mathbf{e}_l).$$

Above, $\vec{\mathfrak{K}}$ gathers all the curvature functions for vertices in V_c, and $\hat{\mathfrak{K}}^l$ denotes its affine approximation near $\mathbf{e}_l$. The Jacobian of $\vec{\mathfrak{K}}$ at $\mathbf{e}_l$ is denoted with $J_{\mathfrak{K}}^l$, and we see that we are merely performing a first-order Taylor expansion of $\vec{\mathfrak{K}}$ at $\mathbf{e}_l$. The explicit formula for $J_{\mathfrak{K}}^l$ is contained within Appendix C.

Thus, the convex subproblem that we solve at each iteration to address the standard setup is the following:

$$\begin{aligned} &\underset{\mathbf{e}\in\mathbb{R}^{|E|}}{\text{minimize}} && E(\mathbf{e}) && && (11) \\ &\text{subject to} && (\sigma_a^i)^2(\mathbf{e}) \le C_a, && i = 1, \dots, |F| && \\ & && (\sigma_b^i)^2(\mathbf{e}) \ge C_b, && i = 1, \dots, |F| && \\ & && K_i^2(\mathbf{e}) \le C_K, && i = 1, \dots, |F| && \\ & && \hat{\mathfrak{K}}^l(\mathbf{e}) = \vec{\mathfrak{K}}^0, && && (12) \end{aligned}$$

where $\vec{\mathfrak{K}}^0$ collects all of our desired curvature values $\mathfrak{K}_j^0$ into a $|V_c| \times 1$ column vector.

Before moving on, we explore the linearization further and note a simplification. Writing out (12) and manipulating it, we see that:

$$\begin{aligned} \hat{\mathfrak{K}}^l(\mathbf{e}) =& \vec{\mathfrak{K}}(\mathbf{e}_l) + J_{\mathfrak{K}}^l(\mathbf{e} - \mathbf{e}_l) = \vec{\mathfrak{K}}^0 \\ \implies& J_{\mathfrak{K}}^l\mathbf{e} = \vec{\mathfrak{K}}^0 - \vec{\mathfrak{K}}(\mathbf{e}_l). \end{aligned} \tag{13}$$

In arriving at (13) we used the fact that $J_{\mathfrak{K}}^l\mathbf{e}_l = 0$. This follows by noting that $\vec{\mathfrak{K}}$ is invariant to scaling: $\vec{\mathfrak{K}}(c\mathbf{e}) = \vec{\mathfrak{K}}(\mathbf{e})$ for any scalar $c > 0$ and $\mathbf{e} \in \mathbb{R}_{>0}^{|E|}$. This makes sense if you note that scaling all the edges of a triangle uniformly leaves the angles invariant. Ultimately, the invariance under scaling implies that the Jacobian of $\vec{\mathfrak{K}}$ at any point $\mathbf{v}$, contains $\mathbf{v}$ within its kernel.

7.2 Additional Details

We note here a few more aspects of SCP and its specialization to our framework. First, we should note that the convex subproblem just mentioned may prove to be infeasible at times. This happens when the bounded distortion constraints (6) - (8) are too strict, and the bounded distortion space does not intersect the linear subspace given by the linearized curvature constraints (12).

Thus, we soften this hard constraint to a soft constraint, and modify the subproblem by removing (12) and replacing (11) with:

$$E(\mathbf{e}) + \lambda_l \left\| J_{\mathfrak{K}}^l\mathbf{e} - \left(\vec{\mathfrak{K}}^0 - \vec{\mathfrak{K}}(\mathbf{e}_l)\right)\right\|_\infty,$$

where λ_l is a parameter that modulates the penalty for violating the original hard constraint. Conversion to the soft constraint guarantees feasibility as the bounded distortion space of discrete metrics is always nonempty with suitable choices of distortion bounds ($C_a \ge C_b$). The parameter λ_l is dynamically set in each iteration with the following formula:

$$\lambda_l = \frac{100\ E(\mathbf{e}_l)}{\left\|\vec{\mathfrak{K}}^0 - \vec{\mathfrak{K}}(\mathbf{e}_l)\right\|_\infty},$$

which attempts to ensure that the soft constraint is weighted at approximately 100 times the value of the energy. This value can obviously be tuned, but we found that this setting works well.

Secondly, we note that after the solution of the subproblem, we do not take the solution right away, but employ a backtracking line search to pick the next point $\mathbf{e}_{l+1}$. For this line search, we prioritize satisfaction of the curvature constraints, and quantify this with $\max_j \left(\left|\mathfrak{K}_j(\mathbf{e}) - \mathfrak{K}_j^0\right|\right)$, the L_∞ norm of the difference between the curvature and the desired values.

More specifically, let $\mathbf{e}_{l+1}^*$ denote the solution to the subproblem. If the L_∞ norm decreases, we accept the step and let $\mathbf{e}_{l+1} = \mathbf{e}_{l+1}^*$. If not, halve the step size and consider again whether the L_∞ norm has decreased. This procedure is iterated until it completes.

Let us highlight the fact that $\mathbf{e}_{l+1}$ will be bounded distortion. This follows from the fact that both $\mathbf{e}_l$ and $\mathbf{e}_{l+1}^*$ are bounded distortion, and the space of bounded distortion metrics is convex. Most applications of SCP would not be able to boast such a property, because their inequality constraints are nonconvex.

Lastly, once a flat metric is obtained or a maximum number of 500 iterations is reached, the final parametrization is obtained by laying the triangles one-by-one in the parametric domain by traversing the dual graph of the mesh as described in [Springborn et al. 2008, Section 3.3].

8 Implementation

We have implemented the algorithm in Matlab, where we utilized the CVX [Grant et al. 2008] modeling system. The underlying SOCP solver was Mosek [ApS 2015]. CVX is a perfect candidate to explore the great flexibility of our approach, as it allows one to formulate convex optimizations like ours through a simple modeling language. The constraints and objectives of any of our variants are expressed as standard Matlab expressions and switching from one type of energy (or constraint) to another, boils down to changing one line of code. In order for CVX to identify the model as convex, the objective function and constraints have to adhere to the Disciplined Convex Programming (DCP) rules. Fortunately, the expressions described in Section 6.2 already obey these rules, which makes implementation rather simple.

8.1 Variants Implemented

With so many variants suggested in Section 6.2, we had to limit ourselves to just a few to show results for. The ones we list below perform well and provide a broad sampling of the possibilities of our framework. The first three variants optimize an isometric measure, aiming to produce a map with as low isometric distortion as possible. The fourth optimizes a conformal measure and prioritizes minimization of conformal distortion. The final variant optimizes a difference energy which projects a reference parametrization to a bounded distortion space of parametrizations. In order to state these variants briefly, we refer back to the framework presented in Section 6.1, and describe the constrained optimization problem that is solved in each variant. The curvature constraints (9) were included in all variants, so this is implicit in the descriptions below.

The first variant (V1) utilized the following energy for (5):

$$\tau(\mathbf{e}) := \max_{\Delta_i} \tau_i(\mathbf{e}).$$

As noted in Section 6.2.3, optimizing for this energy E produces effective bounds on isometric and conformal distortion bounds. Thus the distortion inequalities (6) - (8) were dropped, resulting in greater computational speed.

The second variant (V2) utilized a weighted sum as the energy E:

$$\beta(\mathbf{e}) := \sum_{\Delta_i} w_i \beta_i(\mathbf{e}), \text{ where } w_i = \frac{\text{Area}(\Delta_i)}{\text{Area}(S)}.$$

Note that the weight is a normalized area for Δ_i under the initial discrete metric. As E was a weighted sum, the distortion inequalities were kept (6) - (8) with values $C_a = 20^2, C_b = 1/20^2, C_K = 20^2$.

The third variant (V3) used the Green-Lagrange tensor as the energy:

$$E_{GL}(\mathbf{e}) := \sum_{\Delta_i} w_i E^i_{GL}(\mathbf{e}).$$

It utilized the same distortion inequalities as V2.

The fourth variant (V4) used the following as the energy:

$$E(\mathbf{e}) = \max_{\Delta_i} |\eta_i(\mathbf{e})|.$$

For the distortion inequalities, only (7) was kept with $C_b = 1$. This served dual purposes: it preserved positive-definiteness of the metric (and thus the triangle inequality), and also prevented downward scaling to the zero solution.

The fifth variant (V5) used a difference energy for projection:

$$E_{GLD}(\mathbf{e}) := \sum_{\Delta_i} w_i E^i_{GLD}(\mathbf{e}).$$

For the distortion inequalities, the isometric inequalities (6) and (7) were kept with values $C_a = 3.5^2, C_b = 1/3.5^2$.

8.2 Acceleration with CETM

With many models, our procedure initially moves quickly toward the desired curvature values, needing only a few iterations to roughly approximate them. Further iterations seem to make less significant progress, so we employ CETM [Springborn et al. 2008] as an acceleration technique. More specifically, we consider again L_∞ norm of the difference between the curvature and the desired values: $\max_j \left(\left|\mathfrak{K}_j(\mathbf{e}) - \mathfrak{K}^0_j\right|\right)$, and stop the iterations when this value reaches a threshold. For the results in this paper, we took this threshold value to be 1 degree ($2\pi/360$ radians). When the threshold is reached, CETM is invoked to obtain a completely flat metric. This acceleration step greatly reduces the number of iterations needed, and has little effect on the distortion overall, due to the low threshold value. For specific data, see Section 9.3.

9 Results

The models used to produce our results, along with the field, the cone angles and their positions are all taken, as is, from the benchmark of [Myles et al. 2014]. The statistics of these models (mesh size, number of cones, genus, etc.) and a thorough comparison of different parametrization methods are available for download on the ACM website as supplementary material for [Myles et al. 2014].

We augment the comparison of [Myles et al. 2014] with implementations of the constrained optimization methods of [Aigerman et al. 2014] and the $ARAP - L_\infty$ approach [Levi and Zorin 2014], as well as the conformal CETM method [Springborn et al. 2008]. [Aigerman et al. 2014] was generalized to cone manifolds with the addition of seamless constraints (see [Bommes et al. 2009]), and the frames for optimization were initialized based on the cross field. The cross field from [Bommes et al. 2009] was also used to initialize the $ARAP - L_\infty$ method. CETM and our method have no need for such a field. CETM is utilized for comparison to our conformal variant, and does not accommodate isometric distortion bounds. Finally, we do not compare to the method of [Myles et al. 2014] as unlike all other methods, they perform connectivity changes and allow for additions and removals of cones.

9.1 Discussion of Figures

For all of the figures, conformal distortion is quantified with $k_i \in [0, 1)$. The choice of k_i instead of K_i was made simply because the potential range for values of K_i extends up to ∞ and would be hard to plot. For the figures with isometric distortion comparisons, this is quantified by $\sqrt{\tau_i}$ and the axis in the histograms covers $\sqrt{\tau_i} \in [1, 5]$. For the heat maps, dark blue represents no distortion, $k_i = 0$ or $\tau_i = 1$, and red represents high distortion, $k_i \geq 0.4$ or $\sqrt{\tau_i} \geq 4$.

We turn first to Figure 5 which compares the constrained optimization of [Aigerman et al. 2014] (with the τ energy) with our V1 and V2 variants. The mesh is a topological disk and cones and cuts are omitted to force high distortion and stress test the methods. All three methods are using symmetric isometric energies and are targeted primarily at minimizing isometric distortion (though an effective bound on conformal distortion arises). Comparing V1 with Aigerman's, reveals that the results are nearly identical, which serves as evidence of the optimality and efficacy of both methods

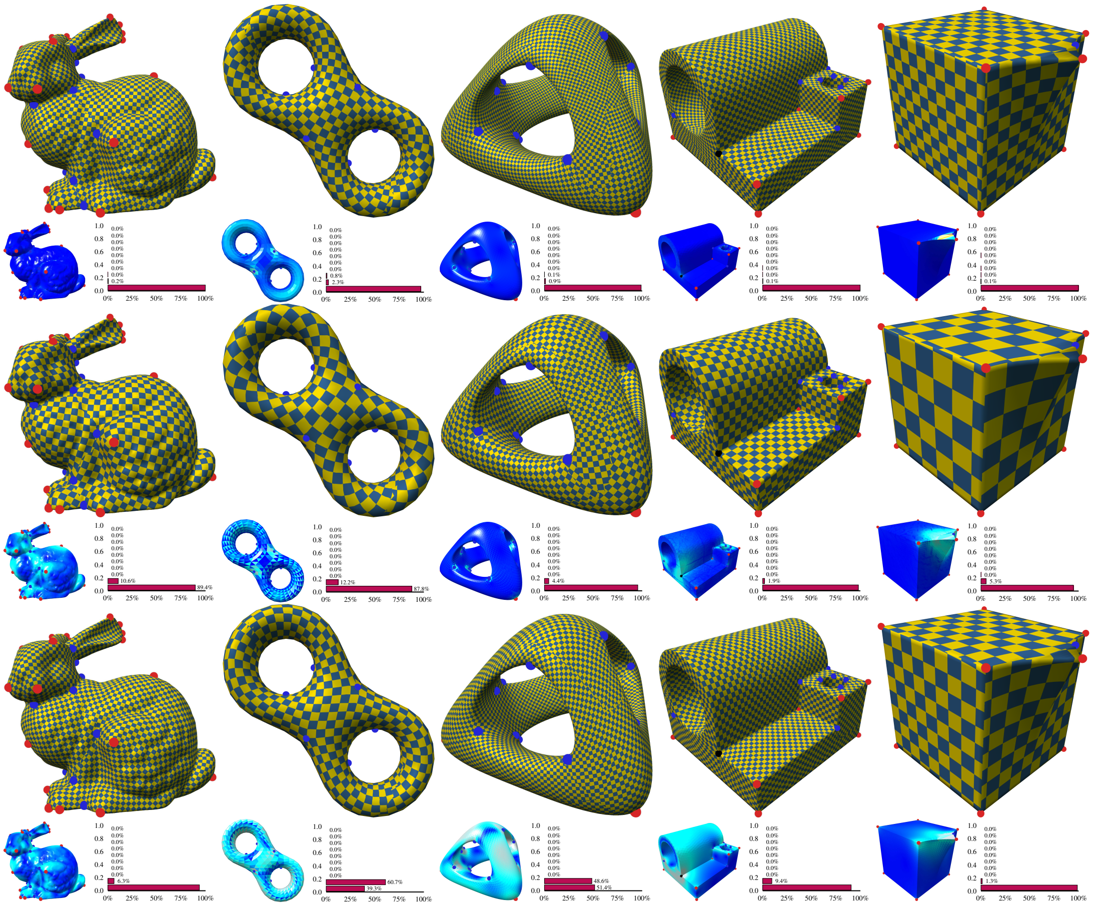


Figure 7: *Conformal maps. Top row: [Springborn et al. 2008]. Middle row: variant V4 of our method, minimizing the maximum of $|\eta|$ over all triangles (Sections 6.2.1 and 8.1). Bottom row: [Lipman 2012] with a bound $k < 0.2$. The histograms and color plots show the conformal distortion k.*

(as they are minimizing equivalent functionals). The result for V2 also has comparable distortion values, albeit with a higher maximum isometric distortion. This sacrifice seems to allow a visually smoother parametrization however, which is oftentimes desirable.

Figure 6 is quite similar, but it also includes results from V3. Again, we have a complicated mesh with disk topology and omit cones and cuts to challenge the methods, and they all produce valid parametrizations in these trying circumstances. We note here that CETM failed to converge on this mesh. V1 and [Aigerman et al. 2014] again produce nearly identical results, advocating for their optimality, and V2 has comparable distortion values with greater variation, but visually smoother results. Finally, we can see that V3 has higher isometric distortion, comparable conformal distortion, and visually smooth results.

Figure 10 presents comparisons on five models, demonstrating results for CETM, $ARAP - L_\infty$, [Aigerman et al. 2014], and V1 (going in order of the rows from top to bottom). These models are not of disc topology and include cone points. We see that comparable results are obtained for the last three methods, while CETM has higher isometric distortion. This is not surprising, given that CETM does not aim to minimize this kind of distortion.

Figure 7 demonstrates the ability of our framework to calculate conformal parametrizations and compares V4 to CETM and [Lipman 2012] over five models. The results are comparable, with CETM producing the highest maximal conformal distortion but also the lowest average conformal distortion.

In Figure 8 we show the application of our method for projecting a reference map to the bounded distortion space. The energy of V5 strives to preserve the nature of the reference map, while pushing the distortion down below a given threshold and ensuring that the map is locally injective. We compare our result with that of [Aigerman and Lipman 2013].

Lastly, we arrive at Figure 1, which shows five difficult models for which CETM, $ARAP - L_\infty$, and [Aigerman et al. 2014] all fail in one way or another. CETM runs into violation of the triangle inequality, $ARAP - L_\infty$ produces parametrizations with inverted faces, and [Aigerman et al. 2014] fails due to infeasibility. V1 succeeds on all of these, and the resulting parametrizations are shown.

9.2 Robustness & Running Times

More generally, we found our methods to be very robust in terms of failure rate, and slightly better than [Aigerman et al. 2014], which seems to have the highest success rate among all the methods we tried (including those methods under examination by [Myles et al. 2014]). On the benchmark of 114 models, we recognized that 7 models (filigree, helmet, raptor, seahorse2, vh-skin, brain, pegaso) had impossible-to-realize cone angles with the given mesh connectivity, and any method that does not alter connectivity is doomed to fail on these models. Of the remaining 107 models, we ran V1

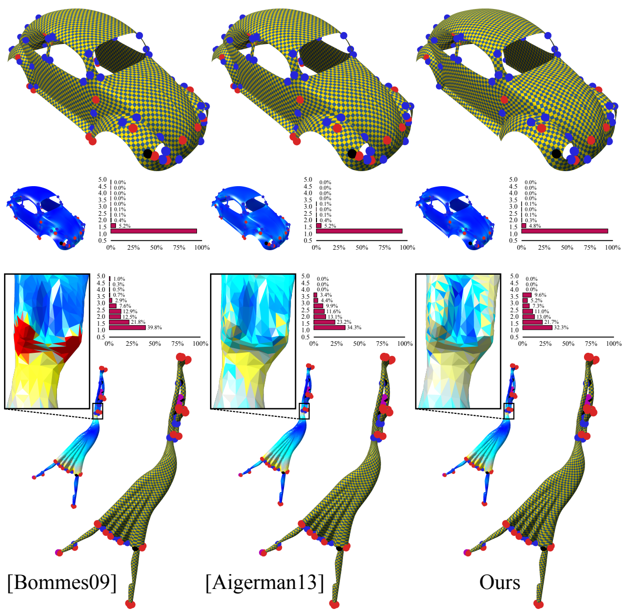


Figure 8: *Projection to the bounded distortion space. The results on the left were computed by [Bommes et al. 2009] and are not locally injective, containing 6 (Beetle) and 88 (Dancer) inverted faces. These parametrizations were projected to the bounded isometric distortion space with $\tau < 3.5^2$ by the method of [Aigerman and Lipman 2013] and variant V5 of our method. Both methods produced maps that are free of inverted elements and adhere to the distortion bound. However, Aigerman's results failed to be locally injective at several vertices, where the angles sums were altered by 2π. Our V5 converged to a metric with the prescribed cone angles, corresponding to a locally injective map.*

of our framework which succeeded on 103 of them, while [Aigerman et al. 2014] succeeded on 97 of them, and $ARAP - L_\infty$ on 93. These are relatively high success rates. For comparison, CETM only produced valid metrics on 18 models (though we haven't tried the edge flips heuristic suggested in [Springborn et al. 2008]). We note also that when [Aigerman et al. 2014] fails (due to infeasibility of the convex program), a parametrization is very hard or impossible to recover. Similar to other metric flattening approaches (e.g. [Ben-Chen et al. 2008; Springborn et al. 2008]), a failure of our method is characterized by the inability of the method to produce the prescribed cone metric up to numerical precision. If a flat metric is obtained, the parametrization is obviously locally injective and has the desired bounds. Moreover, if a flat metric is not obtained, the partially flat metric is still valid and is guaranteed to have bounded distortion. This allows for a quick recovery by feeding the metric to a different method such as [Lévy et al. 2002], which always produces a final layout, at the price of violating the distortion bounds. We emphasise that for all the 103 models mentioned above, our method produced a flat metric and a locally injective map with low bounded distortion.

As for computational efficiency, the running time of each iteration of our algorithm are comparable to those of [Lipman 2012; Aigerman et al. 2014] which all solve second-order cone programs with similar Laplacian-like sparsity patterns. For small mesh sizes of up to 10K triangles, the entire running time of our unoptimized Matlab implementation was typically less than 2 minutes. For the majority of models, less than 15 iterations were required to reduce the curvature, however, many models required larger amount of iterations (Figure 9). For the large models in the data set with $100K$ triangles, the average time for a single iteration was approximately 1 minute.

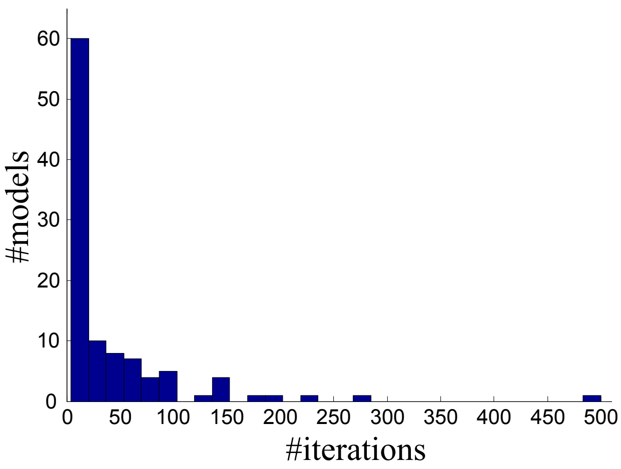


Figure 9: *Histogram of the number of iterations. The average number of iterations for V1 on the 114 models of [Myles et al. 2014] was 43.7 and the median was 15.5. A single model (bozbezbozzel) required the maximal number of iterations (500).*

9.3 CETM Acceleration Effects

We note here that the use of CETM as an acceleration tactic does not drastically affect the distortion or running time, while reducing the number of iterations needed for our method overall. For the 103 models that variant V1 succeeded on, the median and average percentage increases in the distortion were 0.122% and 2.667%, respectively. The average was dragged up by 7 outliers with percentage increases that were above 5%. These outliers may be avoided by decreasing the 1 degree threshold at the expense of additional iterations.

As for running time, for all 103 models, the time spent on CETM accounted for less than 1% of the total running time. For CETM to complete, an average of only 2.5 iterations was needed, demonstrating the propriety of the threshold value of 1 degree.

10 Summary and Discussion

We presented a novel framework for global parametrization which utilizes the edge lengths squared as variables to describe the *convex* space of bounded distortion discrete metrics. With the addition of nonlinear curvature constraints, we obtain a constrained optimization problem that can be used to calculate bounded distortion global parametrizations with prescribed cone metric. The optimization problem is solved with sequential convex programming (SCP), and the simplicity of the problem leads to an effective and highly robust method.

Many popular isometric and conformal measures, including symmetric measures (which are highly desirable) are shown to be convex or quasiconvex functions in our variables. Variants of the basic method are used to calculate maps that strives to be as isometric as possible, as conformal as possible, or project a reference map with special nature to the bounded distortion space. With the results in the paper, we show that our method matches the state-of-the-art in terms of quality and robustness, while introducing a refreshing approach to a hard problem.

10.1 Limitations and Future Work

Within our work, we have not discussed several parametrization constraints that may be desirable; such as positional constraints, cross-field alignment, alignment to sharp features, and prescription of holonomy angles. The last two types may be easily implemented with additional constraints over sum of angles of a type similar to those already implemented. The positional constraints and cross-field alignment are more difficult and would require more complicated nonconvex constraints, and are not the main point of concern with our approach. We note also that the output of our method may be fed as input into other methods which are more suited to achieving these aims. Further work should clarify the strategies used to achieve desirable parametrization constraints.

The observed convergence rate of the method was that the curvature is reduced significantly after relatively low number of iterations, while for some models it takes many iterations to reach a completely flat metric. We view this as a limitation and feel that future research and better design of optimization procedures will allow for great improvements in the practical performance of the method and its variants.

As a promising direction for further research, we would like to incorporate into the flattening process, an automatic way to insert cone singularities at strategic positions, which will allow for even lower bounds on the distortion to be realized.

Finally, we view as an exciting avenue of future exploration, the possibility to compute bounded distortion maps of tetrahedral meshes. The same philosophy in this paper should generalize in light of Theorems 1 and 2.

Acknowledgements

This research was partially funded by the Israel Science Foundation (grants No. 1869/15 and 2102/15). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPU.

References

AIGERMAN, N., AND LIPMAN, Y. 2013. Injective and bounded distortion mappings in 3D. *ACM Transactions on Graphics (TOG) 32*, 4, 106.

AIGERMAN, N., PORANNE, R., AND LIPMAN, Y. 2014. Lifted bijections for low distortion surface mappings. *ACM Transactions on Graphics (TOG) 33*, 4, 69.

APS, M. 2015. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28).*

BEN-CHEN, M., GOTSMAN, C., AND BUNIN, G. 2008. Conformal flattening by curvature prescription and metric scaling. *Computer Graphics Forum 27*, 2, 449–458.

BOBENKO, A. I., PINKALL, U., AND SPRINGBORN, B. A. 2015. Discrete conformal maps and ideal hyperbolic polyhedra. *Geometry & Topology 19*, 4, 2155–2215.

BOMMES, D., ZIMMER, H., AND KOBBELT, L. 2009. Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3, 77.

BOMMES, D., CAMPEN, M., EBKE, H.-C., ALLIEZ, P., AND KOBBELT, L. 2013. Integer-grid maps for reliable quad meshing. *ACM Transactions on Graphics (TOG) 32*, 4, 98.

BOMMES, D., LÉVY, B., PIETRONI, N., PUPPO, E., SILVA, C., TARINI, M., AND ZORIN, D. 2013. Quad-mesh generation and processing: A survey. *Computer Graphics Forum 32*, 6, 51–76.

BOTSCH, M., KOBBELT, L., PAULY, M., ALLIEZ, P., AND LÉVY, B. 2010. *Polygon mesh processing*. CRC press.

BOYD, S., AND VANDENBERGHE, L. 2004. *Convex optimization*. Cambridge university press.

CAMPEN, M., BOMMES, D., AND KOBBELT, L. 2015. Quantized global parametrization. *ACM Transactions on Graphics (TOG) 34*, 6, 192.

CHEN, R., AND WEBER, O. 2015. Bounded distortion harmonic mappings in the plane. *ACM Transactions on Graphics (TOG) 34*, 4, 73.

CHEN, R., WEBER, O., KEREN, D., AND BEN-CHEN, M. 2013. Planar shape interpolation with bounded distortion. *ACM Transactions on Graphics (TOG) 32*, 4, 108.

CHIEN, E., CHEN, R., AND WEBER, O. 2016. Bounded distortion harmonic shape interpolation. *ACM TOG 35*, 4.

DINH, Q. T., AND DIEHL, M. 2010. Local convergence of sequential convex programming for nonconvex optimization. In *Recent Advances in Optimization and its Applications in Engineering*. Springer, 93–102.

EPPSTEIN, D. 2005. Quasiconvex programming. *Combinatorial and Computational Geometry 52*, 287–331.

FLOATER, M., AND HORMANN, K. 2005. Surface Parameterization: a Tutorial and Survey. *Advances In Multiresolution For Geometric Modelling*.

GRANT, M., BOYD, S., AND YE, Y., 2008. CVX: Matlab software for disciplined convex programming.

HORMANN, K., LÉVY, B., AND SHEFFER, A. 2007. Mesh parameterization: Theory and practice. *SIGGRAPH Course Notes*.

KÄLBERER, F., NIESER, M., AND POLTHIER, K. 2007. QuadCover: Surface Parameterization using Branched Coverings. *Computer Graphics Forum 26*, 3, 375–384.

KHAREVYCH, L., SPRINGBORN, B., AND SCHRÖDER, P. 2006. Discrete conformal mappings via circle patterns. *ACM Trans. Graph. 25*, 2, 412–438.

KOVALSKY, S. Z., AIGERMAN, N., BASRI, R., AND LIPMAN, Y. 2014. Controlling singular values with semidefinite programming. *ACM Transactions on Graphics (TOG) 33*, 4, Article 68.

KOVALSKY, S. Z., AIGERMAN, N., BASRI, R., AND LIPMAN, Y. 2015. Large-scale bounded distortion mappings. *ACM Transactions on Graphics (TOG) 34*, 6, 191.

LAI, Y.-K., JIN, M., XIE, X., HE, Y., PALACIOS, J., ZHANG, E., HU, S.-M., AND GU, X. 2010. Metric-driven rosy field design and remeshing. *Visualization and Computer Graphics, IEEE Transactions on 16*, 1, 95–108.

LEE, J. M. 1997. *Riemannian Manifolds: An Introduction to Curvature*. Springer.

LEVI, Z., AND ZORIN, D. 2014. Strict minimizers for geometric optimization. *ACM Transactions on Graphics (TOG) 33*, 6, 185.

LÉVY, B., PETITJEAN, S., RAY, N., AND MAILLOT, J. 2002. Least squares conformal maps for automatic texture atlas generation. *ACM Transactions on Graphics (TOG) 21*, 3, 362–371.

LIPMAN, Y. 2012. Bounded distortion mapping spaces for triangular meshes. *ACM Transactions on Graphics (TOG) 31*, 4, 108.

LIU, L., ZHANG, L., XU, Y., GOTSMAN, C., AND GORTLER, S. 2008. A local/global approach to mesh parameterization. *Computer Graphics Forum 27*, 5, 1495–1504.

LIU, T., GAO, M., ZHU, L., SIFAKIS, E., AND KAVAN, L. 2016. Fast and robust inversion-free shape manipulation. *Computer Graphics Forum 35*, 2.

MAILLOT, J., YAHIA, H., AND VERROUST, A. 1993. Interactive texture mapping. *Proceedings of SIGGRAPH 1993*, 27–34.

MYLES, A., AND ZORIN, D. 2012. Global parametrization by incremental flattening. *ACM Trans. Graph. 31*, 4 (July), 109:1–109:11.

MYLES, A., AND ZORIN, D. 2013. Controlled-distortion constrained global parametrization. *ACM Trans. Graph. 32*, 4 (July), 105:1–105:14.

MYLES, A., PIETRONI, N., AND ZORIN, D. 2014. Robust field-aligned global parametrization. *ACM Trans. Graph. 33*, 4 (July), 135:1–135:14.

PORANNE, R., AND LIPMAN, Y. 2014. Provably good planar mappings. *ACM Transactions on Graphics (TOG) 33*, 4, 76.

SCHREINER, J., ASIRVATHAM, A., PRAUN, E., AND HOPPE, H. 2004. Inter-surface mapping. *ACM Transactions on Graphics (TOG) 23*, 3, 870–877.

SCHÜLLER, C., KAVAN, L., PANOZZO, D., AND SORKINE-HORNUNG, O. 2013. Locally injective mappings. In *Computer Graphics Forum*, vol. 32, Wiley Online Library, 125–135.

SHEFFER, A., AND DE STURLER, E. 2001. Parameterization of faceted surfaces for meshing using angle-based flattening. *Engineering with Computers 17*, 3, 326–337.

SORKINE, O., COHEN-OR, D., GOLDENTHAL, R., AND LISCHINSKI, D. 2002. Bounded-distortion piecewise mesh parameterization. In *Proceedings of IEEE Visualization*, IEEE Computer Society, 355–362.

SPRINGBORN, B., SCHRÖDER, P., AND PINKALL, U. 2008. Conformal equivalence of triangle meshes. *ACM Transactions on Graphics (TOG) 27*, 3, 77.

VAXMAN, A., CAMPEN, M., DIAMANTI, O., PANOZZO, D., BOMMES, D., HILDEBRANDT, K., AND BEN-CHEN, M. 2016. Directional field synthesis, design, and processing - state of the art report. *Computer Graphics Forum 35*, 2.

WEBER, O., AND ZORIN, D. 2014. Locally injective parametrization with arbitrary fixed boundaries. *ACM Transactions on Graphics (TOG) 33*, 4, 75.

WEBER, O., MYLES, A., AND ZORIN, D. 2012. Computing extremal quasiconformal maps. *Computer Graphics Forum 31*, 5, 1679–1689.

A Expressions for η_i

The same notation from Section 4.3 is used, and these are derived with the choice of charts mentioned in Section 4.1.

$$\mathrm{Re}\,(\eta_i) = \left(\frac{a_0 b_0 + a_0 c_0 - (b_0 - c_0)^2}{2a_0 \mathscr{T}^2}\right) a_1 + \left(\frac{-a_0 + b_0 - c_0}{2\mathscr{T}^2}\right) b_1 + \left(\frac{-a_0 - b_0 + c_0}{2\mathscr{T}^2}\right) c_1$$

$$\mathrm{Im}\,(\eta_i) = \left(\frac{b_0 - c_0}{2a_0 \mathscr{T}}\right) a_1 + \left(\frac{-1}{2\mathscr{T}}\right) b_1 + \left(\frac{1}{2\mathscr{T}}\right) c_1.$$

As before, note the linearity of the expressions in the edge lengths squared of $f(\Delta_i)$ (the induced edge lengths squared).

B Quasiconvexity of ARAP and LSCM

For the sake of brevity, we merely sketch the proofs of quasiconvexity of E^i_{ARAP} and E^i_{LSCM}. The basic approach is the same for both. Let us first express both of these quantities in terms of our usual metric tensor coordinates.

$$\begin{aligned} E^i_{ARAP} &= \frac{1}{2}\left((\sigma^i_a - 1)^2 + (\sigma^i_b - 1)^2\right) \\ &= \mathcal{A}_i - \sqrt{\mathcal{A}_i + 2|\eta_i|} - \sqrt{\mathcal{A}_i - 2|\eta_i|} + 1 \end{aligned} \tag{14}$$

$$\begin{aligned} E^i_{LSCM} &= \frac{1}{2}\left((\sigma^i_a)^2 + (\sigma^i_b)^2\right) - \sigma^i_a \sigma^i_b \\ &= \mathcal{A}_i - \sqrt{\mathcal{A}_i^2 - 4|\eta_i|^2}. \end{aligned} \tag{15}$$

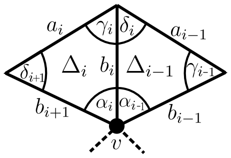
We get the initial equality in (14) from [Liu et al. 2008]. As can be seen by the formulae, bounding these quantities from above results in sublevel sets that are radially symmetric about the $\mathcal{A}_i$ axis. Our goal is to show that these sets are convex. As in Figure 4, we gain intuition by considering the cross sections in the $\mathcal{A}_i$ and $|\eta_i|$ plane (they may be easily graphed, but we omit these for brevity).

Noting the transition from cylindrical coordinates to Euclidean coordinates, it is not hard to see that convexity for a radially symmetric region with cross section R_{cs} is equivalent to convexity of R_{cs} plus the following condition:

$$R_{cs} = \hat{R}_{cs} := \{(|\eta_i|, \mathcal{A}_i) \mid |\eta_i| \leq x_0 \text{ for some } (x_0, \mathcal{A}_i) \in R_{cs}\}.$$

The cross sections for the sublevel sets of E^i_{ARAP} and E^i_{LSCM} may be shown to be convex by showing that they are the hypographs of concave functions expressing the boundaries of these cross sections (these graphs have $\mathcal{A}_i$ as the independent variable and $|\eta_i|$ as the dependent variable).

C Jacobian of Curvature Functions



The entries of the Jacobian consist of the partial derivatives of $\mathfrak{K}_j$ by edge lengths squared. To deter confusion, we note that the indices i, j chosen here are independent of those in Section 7.1. This matrix will be sparse, with the curvature at a vertex only depending on the lengths of the edges in its 1-ring. To set notation, we again utilize the figure to the right. In addition, the following are helpful:

$$\mathrm{Area}(\Delta_i) = \frac{1}{4}\sqrt{2a_i b_i + 2a_i b_{i+1} + 2b_i b_{i+1} - a_i^2 - b_i^2 - b_{i+1}^2}$$

$$\mathrm{Area}(\Delta_{i-1}) = \frac{1}{4}\sqrt{2a_{i-1} b_i + 2a_{i-1} b_{i-1} + 2b_i b_{i-1} - a_{i-1}^2 - b_i^2 - b_{i-1}^2}$$

where the areas are in terms of the induced metric (not the original). With $\mathfrak{K}_j$ denoting curvature at v, some calculation reveals the following formulae:

$$\frac{\partial \mathfrak{K}_j}{\partial a_i} = -\frac{1}{4\mathrm{Area}(\Delta_i)}$$

$$\frac{\partial \mathfrak{K}_j}{\partial b_i} = \frac{1}{2b_i}\left(\frac{a_i + b_i - b_{i+1}}{4\mathrm{Area}(\Delta_i)} + \frac{a_{i-1} + b_i - b_{i-1}}{4\mathrm{Area}(\Delta_{i-1})}\right).$$

The above expressions are suitable for implementation, but the second equation may be simplified for additional geometric insight:

$$\frac{\partial \mathfrak{K}_j}{\partial b_i} = \frac{\cot(\gamma_i) + \cot(\delta_i)}{2b_i}.$$

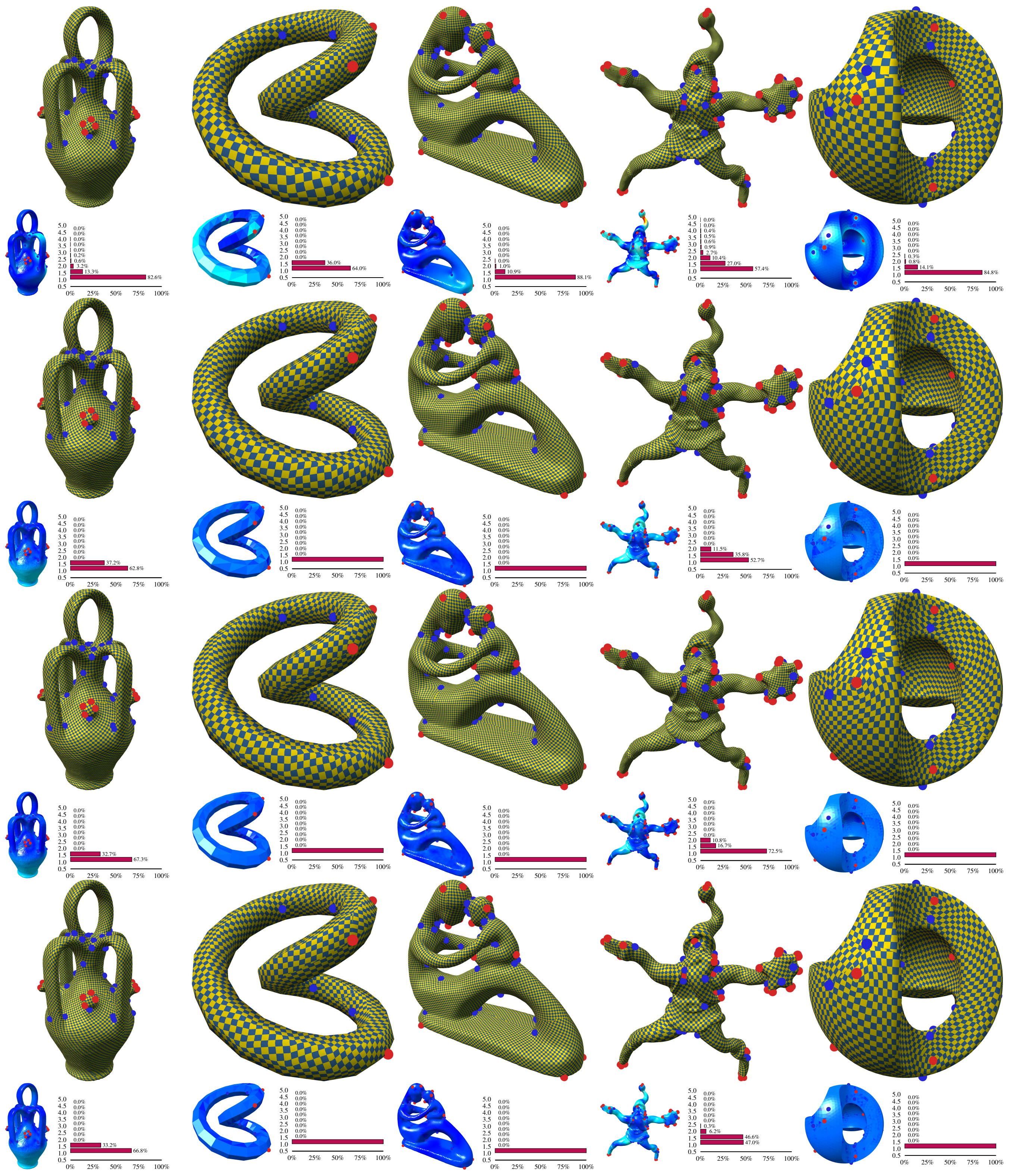


Figure 10: *Comparison of different methods on five models from [Myles et al. 2014]. The cone positions and angles, as well as the field, were generated with [Bommes et al. 2009]. Top to bottom. [Springborn et al. 2008],* $ARAP - L_\infty$ *[Levi and Zorin 2014], [Aigerman et al. 2014], and variant V1 of our method. The histograms and color visualization refer to the isometric distortion* $\sqrt{\tau} = \max(\sigma_a, 1/\sigma_b)$*. All four methods produced locally injective parametrizations. While the conformal method (top) resulted in higher maximum and average isometric distortion, the three other methods produced similar distribution of the distortion.*