# Strict Minimizers For Geometric Optimization

Zohar Levi *
New York University

Denis Zorin †
New York University

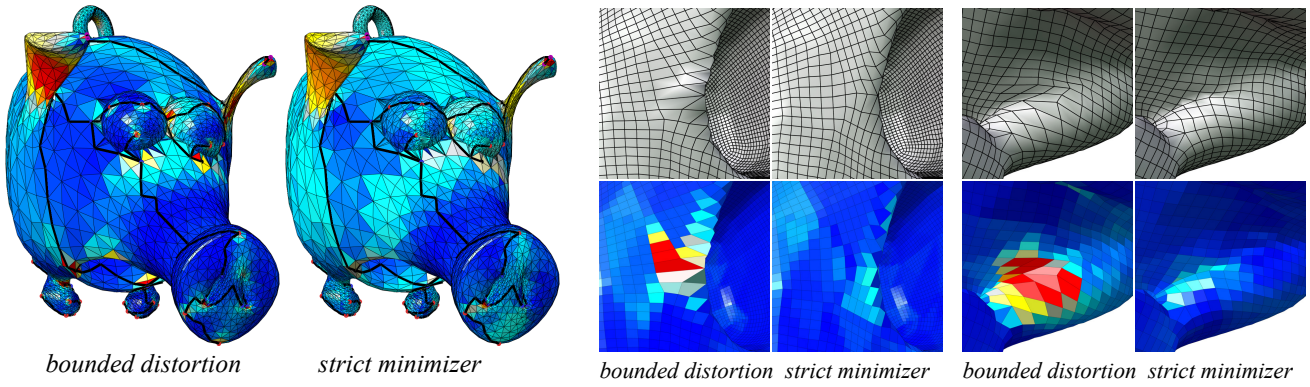



**Figure 1:** *Comparing the parametrization distortion of [Aigerman and Lipman 2013], marked* bounded distortion*, and our strict minimizer. The distortion is color-coded on the facets of the source mesh: dark blue corresponds to distortion 0 and red to distortion above* 0.6*. The two sets of magnified images show parts of the mesh after quadrangulation using two parametrizations, with the same scale for distortion.*

## Abstract

We introduce the idea of *strict minimizers* for geometric distortion measures used in shape interpolation, deformation, parametrization, and other applications involving geometric mappings. The $L_\infty$-norm ensures the tightest possible control on the worst-case distortion. Unfortunately, it does not yield a unique solution and does not distinguish between solutions with high or low distortion below the maximum. The strict minimizer is a minimal $L_\infty$-norm solution, which always prioritizes higher distortion reduction. We propose practical algorithms for computing strict minimizers. We also offer an efficient algorithm for $L_\infty$ optimization based on the ARAP energy. This algorithm can be used on its own or as a building block for an ARAP strict minimizer. We demonstrate that these algorithms lead to significant improvements in quality.

**CR Categories:** I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—[Geometric algorithms, languages, and systems];

**Keywords:** geometric modeling, shape interpolation, deformation, parametrization

**Links:** DL PDF

*e-mail: zohar@cs.nyu.edu
†e-mail: dzorin@cs.nyu.edu

**ACM Reference Format**
Levi, Z., Zorin, D. 2014. Strict Minimizers For Geometric Optimization. ACM Trans. Graph. 33, 6, Article 185 (November 2014), 14 pages. DOI = 10.1145/2661229.2661258 http://doi.acm.org/10.1145/2661229.2661258.


0730-0301/14/11-ART185 $15.00.
DOI: http://dx.doi.org/10.1145/2661229.2661258

## 1 Introduction

A variety of problems in geometry processing are solved by minimizing a global measure of distortion over a surface or volume. For example, computing the deformation of an object to a new pose can be achieved by setting a few target locations to new positions and minimizing a measure of isometric distortion; parameterizing a surface, or computing surface-to-surface map via an intermediate domain may be done by computing a pair of maps minimizing deviation from conformality; fitting a high-order approximation of a surface to a mesh optimizes the difference between point positions and normals at corresponding points.

In most cases, the global functional is obtained by making two choices:

(1) *a local*, pointwise measure of distortion (for example, deviation from isometry or quasiconformal distortion);

(2) a way to aggregate pointwise distortion into a single scalar measuring global distortion.

The second choice determines how distortion is distributed on the surface. $L_2$-norm of the distortion, viewed as a function on the surface, is popular, as it often leads to the most computationally efficient methods. It emphasizes the decrease in average distortion, but allows arbitrarily high local distortion. This property leads to significant problems for most applications: under- and oversampled areas in quadrangulation and remeshing, low accuracy in simulations using meshes with high distortion, and visual artifacts in texture mapping.

On the opposite end of the spectrum, $L_\infty$-norm ensures the tightest possible control on the worst-case distortion. Unfortunately, it does not, in general, yield a unique solution. Moreover, below the maximal distortion, it does not distinguish between solutions with just one or all elements (triangles or tets) with high distortion. The specific solution chosen depends on the choice of the solver.

It is also possible to cast the problem of distortion control as *constrained optimization*, e.g., minimizing the $L_2$ norm, but subject to user-defined upper bound on distortion. However, the choice of the bound is constrained by a small number of highest-distortion locations, and does not prevent distortion concentration, typical for $L_2$-norm, as long as it remains below the bound.

In this paper we take a different approach of controlling distortion distribution. We use the notion of *strict minimizers*; these minimizers do not correspond to a globally defined energy. Rather, they are minimal with respect to an ordering of pointwise distortion distributions on a surface. More specifically, we consider an ordering that prioritizes reducing higher distortion over reducing lower distortion (a precise definition is given in Section 4). A strict minimizer minimizes the $L_\infty$-norm of the distortion, while defining the distortion uniquely on each element for an important class of convex distortion measures. Experimentally, strict minimizers lead to substantial improvements in high distortion areas, at the expense of small increases of pointwise distortion spread over large low-distortion areas. Aside from being unique, they are also known to be the limits of $L_p$-norm minimizers, which suggests a degree of regularity.

Computing strict minimizers using a precise algorithm that follows the definition can be impractically expensive. We introduce two essential components for fast computation of strict minimizers: a simple, efficient formulation for $L_\infty$ optimization for certain types of distortion measures; and a relaxation-based algorithm which finds an approximation to the strict minimizer orders of magnitude faster. Our efficient formulation for $L_\infty$ optimization can also be used independently from strict minimization.

While most of our examples are related to parametrization, the algorithm is applicable to other applications, such as deformation, surface-to-surface mapping, and shape interpolation. We compare to a number of previously proposed approaches for controlling distortion and explore the behavior of the proposed method for a number of examples.

## 2 Related Work

**Strict minimizers.** The idea of strict minimizers, defined precisely in Section 4, first appeared in [Rice 1963], as *strict uniform best approximation*. Rice [1963] develops the idea of a unique (in a sense, best) solution to the minimax approximation problem $\min_{a_i} \|f - \sum_i a_i g_i\|_\infty$, where $f$ is a function, defined on a finite set of points; $g_i$ are the approximation basis functions; and $a_i$ are the coefficients to find. Marano [1990] has generalized the properties of strict approximations to convex sets in $\mathbb{R}^n$.

This problem is equivalent to the problem of identifying a best max-norm solution of an overdetermined linear system (*strict Chebyshev solution*, e.g., [Abdelmalek 1977]). Descloux [1963] proves that it is the limit of the $p$-approximations, for $p \to \infty$. Computing a strict approximation as a limit of $p$-norm approximations is often referred to as the *Polya algorithm*. In [Mémoli et al. 2006], the 2-norm of the differential of the map between two surfaces is used as the distortion measure at each point, and the map is defined as the limit of $L_p$ norms of this measure. Behringer [1981] describes a method of computing strict approximations using a simplex-type method.

Strict approximation were considered as a way of aggregating costs or error measures in many domains, e.g., resource allocation [Luss and Smith 1986; Marchi and Oviedo 1992], network bandwidth optimization [Ogryczak and Sliwinski 2007], and filter design.

Finally, we note that Delaunay triangulations are known to find the strict minimizer of the problem of minimizing the maximal angle of a triangulation for a point set. The classical flip algorithm computing Delaunay triangulations [Lawson 1972], by doing local minimization structurally, resembles the relaxation-based algorithm that we propose, with angle magnitude as the distortion measure.

**Local distortion measures.** Optimizing a measure of distortion is the most common approach to shape deformation, parametrization, shape-to-shape mappings and shape interpolation. A variety of local distortion measures have been proposed; in this paper we consider *field-deviation*, measuring alignment of mapping gradients with prescribed directions (e.g. [Kälberer et al. 2007; Ray et al. 2006; Bommes et al. 2009]), which is also a standard component for local-global approaches to minimizing deviation from isometry measured by the 2D and 3D ARAP functional [Alexa et al. 2000; Sorkine and Alexa 2007; Liu et al. 2008; Chao et al. 2010]. More recently, Aigerman and Lipman [2013] adopted from [Sorkine et al. 2002] a different measure of deviation from isometry, which can also be used with our algorithm. In the context of parametrization and deformation, many papers approximate conformal maps [Sheffer and de Sturler 2001; Kharevych et al. 2006; Ben-Chen et al. 2008; Springborn et al. 2008], reducing conformal distortion to zero in the smooth case. Closely related, harmonic maps [Lévy et al. 2002; Desbrun et al. 2002] use the map's gradient norm as a distortion measure. They can also be viewed as minimizing the distance to conformality in cases where boundary conditions do not allow conformal map existence. A different scale-invariant measure of distance to conformality (dilatation or quasi-conformal distortion) is used in [Lipman 2012] and [Weber et al. 2012], which is closely related to the MIPS energy [Hormann and Greiner 1999]. In [Solomon et al. 2011], the distance to a Killing field is used for deformation optimization.

**Global distortion optimization.** With the exception of conformal maps, most of the geometry processing literature uses $L_2$ norms of local distortion measures, although $L_1$ norms, which lead to sparse solutions, are gaining popularity for some applications. General $L_p$-norms are mentioned in [Bommes et al. 2013] as a possible option for global parametrization. Weber et al. [2012] indirectly solve the $L_\infty$ minimization problem by using special properties of the extremal quasiconformal maps, in particular, the fact that in the smooth case, these are unique and characterized by constant distortion measure. $L_\infty$-norm also has a long history in spline theory, typically considered in settings when minimizers are unique.

Constrained formulations are becoming increasingly popular for deformations and parametrization, allowing one to combine an upper bound for local distortion with minimizing a global norm. For example, a constrained field-deviation formulation was considered in [Bommes et al. 2013], with a constraint ensuring bijectivity, but not bounding any specific type of distortion. Lipman [2012] proposes a framework in which any convex global energy, in particular field-deviation (ARAP's global step), can be combined with a bound on QC distortion. This approach is extended to a broader class of distortion bounds in [Aigerman and Lipman 2013]. Schueller et al. [2013] use a penalty method to ensure bijectivity of deformations.

For surface parametrization, distortion distribution is typically strongly affected by the choice of singularities of the parameterization and, in the case of field deviation measure, by the choice of the guiding field. In this paper, we do not consider the problem of choosing singularities or finding positions of singularities; these questions were addressed in many previous papers, e.g., [Kälberer et al. 2007; Ben-Chen et al. 2008; Bommes et al. 2009; Myles and Zorin 2012; Myles and Zorin 2013]. In particular, Myles and Zorin [2012] use a form of ARAP distortion, restricted to conformal maps, to determine singularity placement. Our approach can be used with any fields, but naturally the best possible distortion achievable is strongly affected by the choice of singularities.

## 3 Motivation and Overview

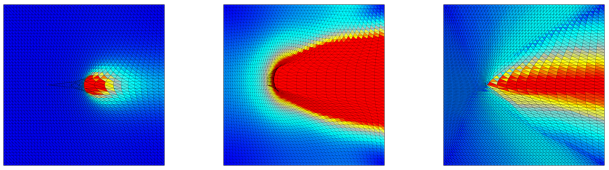
**Notation.** Although our primary focus is on triangular meshes (embedded in $\mathbb{R}^2$ or $\mathbb{R}^3$), most of the considerations in this paper apply also to tetrahedral meshes. Consider a mesh $\mathcal{M}$ with vertices $v_1, \ldots, v_M \in \mathbb{R}^n$ and *elements* (triangles or tetrahedra), $T_1, \ldots T_N$. We assume that a piecewise linear mapping $f : \mathcal{M} \to \mathbb{R}^n$, $n = 2, 3$, is given by its values at vertices $f(v_j) \in \mathbb{R}^n$, $j = 1 \ldots M$, forming a vector $\mathbf{f} \in \mathbb{R}^{nM}$. We denote the constant differential of the map $f$ on an element $T_i$ by $J_i \in \mathbb{R}^{n \times n}$; $J_i = J_i(\mathbf{f}) : \mathbb{R}^{nM} \to \mathbb{R}^{n \times n}$ is a linear function of $\mathbf{f}$ that returns

the Jacobian. All the per-element distortion measures $D_i$ that we consider in this paper are functions of $J_i$.

We start with an informal description of the approach in a specific case, to identify potential problems and possible solutions. Consider the following 2D deformation problem: Given a planar mesh $\mathcal{M}$ with target locations $q_m \in \mathbb{R}^2$ for several vertices $p_m \in \mathcal{M} \subset \mathbb{R}^2$, we would like to compute a piecewise linear deformation $f$, such that $f(p_m) = q_m$. The local distortion measure on each triangle is the deviation of the Jacobian $J_i \in \mathbb{R}^{2\times 2}$ from identity: $D_i(\mathbf{q}) = \|J_i(\mathbf{q}) - I\|_F$, where $\mathbf{q} \in \mathbb{R}^{2M}$ is the vector of the new positions of the vertices.

Suppose we can compute an optimal deformation $f$ with respect to $L_2$ and $L_\infty$ norms of the vector of distortions $[D_1, D_2 \dots D_N]$. (We use the notation $L_p$-norm, rather than $p$-norm, to indicate that the vector is regarded as values of a piecewise-constant function on $\mathcal{M}$, so components' contributions, in the case of $p < \infty$, are weighted by areas of triangles.)

Figure 2, left, shows that for $L_2$-norm, we get a high distortion concentrated near a fixed point. $L_\infty$-norm optimization yields a better result for worst-case triangles, but still allows high distortion concentration. We call this initial $L_\infty$ solution $\mathbf{q}_1$.



$L_2$ $L_\infty$ *strict*

**Figure 2:** *The central point of a square with a fixed boundary is displaced horizontally. Deformations optimal with respect to $L_2$, $L_\infty$ norms, and a strict minimizer are shown.*

A natural idea is to fix a set of triangles $I_1$ for which the maximal distortion is reached: $D_i(\mathbf{q}) = \|D(\mathbf{q})\|_\infty = k_1$, for $i \in I_1$, and attempt to reduce the distortion on the remaining ones as much as possible. More specifically, we solve an $L_\infty$ optimization problem on the subset of triangles $I \setminus I_1$, while imposing the constraint $D_i(\mathbf{q}) \le k_1$ on $I_1$ (this problem is feasible because the solution of the initial $L_\infty$ problem satisfies the constraints). Similarly to the first step, this yields a second set of triangles $I_2$, which reach the second optimal bound $k_2$. Continuing this process, we eventually fix bounds for all triangles, decomposing them into groups $I_1$, $I_2$, $\dots I_L$, with the same distortion. This process allows one to choose one particular solution out of many possible $L_\infty$ minimizers. This solution tries to make the distortion as low as possible everywhere, but always prioritizes reducing high distortion.

This idea leads to several important questions:

1. Is there a description of the result produced by this procedure independent of the algorithm, so that efficient approximations can be designed?
2. $L_\infty$ minimizers are not unique, but they determine sets $I_i$, so the described algorithm may potentially produce different results, depending on the way minimizers are defined. Which one do we choose?
3. As described, the procedure requires $L_\infty$ optimization at every step, and the number of steps may potentially be high: as few as one triangle may be fixed at each step. How can $L_\infty$ optimization be performed efficiently and the number of steps decreased?

In Section 4, we consider the first two questions, introducing the concept of *strict minimizers* and its relation to the algorithm above. In Section 5, we describe an efficient $L_\infty$ minimizer, which is an essential component for an approximate *direct* strict optimization algorithm. Building on the definition of the strict minimizer in the next section, we describe a relaxation-based approximate algorithm that yields comparable results in a fraction of the time.

## 4 Strict Minimizers

In this section, we describe the theoretical foundations of our approach. We start with a general definition, applicable both in the smooth and discrete cases, and then specialize it to the discrete case. We assume that for a map $f : M \to R^n$, defined on a domain $M$ of matching dimension $n = 2$ or $n = 3$, a local pointwise distortion measure $D[f](x) \ge 0$ is chosen. We discuss a number of specific examples of such measures in Section 7. The primary example considered in this paper is *frame field deviation*, commonly used in deformation, feature-aligned parametrization, quadrangulation, and hex meshing in 3D.

Suppose a local coordinate frame of $n$ vectors $e_i$, $n = 2, 3$, is chosen at every point $x$ on $M$. Let $J[f]$ be the matrix representing the differential of $f$ with respect to this frame, and $R$ be an orthonormal matrix with columns corresponding to unit vectors that are viewed as target images for vectors $e_i$. Then the field deviation is defined as

$$D^{FD}[f](x) = \|J[f](x) - R(x)\|_F$$

where $\|\cdot\|_F$ is the Frobenius norm.

**Strict minimizers.** We use an approach to choose a particular $L_\infty$ minimizer, which we demonstrate to be unique in the discrete case, based on the idea of introducing an *ordering* on distortion distributions $D[\mathbf{f}](x)$. While this idea has been applied in many domains, to the best of our knowledge, this idea has not been explored in the geometry processing context.

For any distortion level $z \ge 0$ and mapping $f$, define $H_{f,z} \subset M$ to be the set of points $x$ of $M$ where $D[\mathbf{f}](x) \ge z$, and let $|H_{f,z}|$ be the area/volume of $H_{f,z}$. Then we say that $f$ is *less distorting* than $g$, $f \prec g$, if there is $z_0$ such that $|H_{f,z_0}| < |H_{g,z_0}|$, and for any $z > z_0$, $|H_{f,z}| = |H_{g,z}|$.

If for $f$ and $g$, $|H_{f,z}| = |H_{g,z}|$ for any $z$, then we consider them equivalent ($f \preceq g$, if $f \prec g$, or $f \equiv g$). The relation $\preceq$ defines a total order on the set of equivalence classes of maps.

The idea behind this ordering is that we consider reducing the area of higher distortion always a greater priority than reducing the area for any lower distortion.

**Definition 1.** *We say that $f^*$ has* strictly minimal distortion *or is a* strict minimizer *if $f^* \preceq g$ for any $g$.*

The term originates in [Rice 1962], where a related problem of strict approximation is considered. The definition above applies both to the smooth surface case and to meshes. For meshes, $D[\mathbf{f}]$ is typically piecewise constant per element, so we can represent it as a vector of distortion values and define $|H_{f,z}|$ more explicitly as a sum of element areas. Next, we explain a conceptual algorithm for constructing a strict minimizer. While not practical for any but small meshes, it provides an important reference point and can be used to establish that any two minimizers like this have exactly the same distortion on every element (see Appendix B).

**Discrete case.** Definition 1 applies to meshes directly, as long as the distortion measure can be computed almost everywhere (typically at all interior points of triangles), but it is useful to rephrase it in a more explicit form.

For 2D and 3D meshes and piecewise linear maps, we denote $D[\mathbf{f}] = [D_1[\mathbf{f}], \dots D_N[\mathbf{f}]]$, where $D_i[\mathbf{f}]$ is the distortion on element $i$. The set $H_{f,z}$ in the discrete case is the set of elements with distortion level $z$ or higher. As distortion levels $z$ now form a discrete set $z_0 > \dots > z_L$ for a given $f$ (there is at most as

many distortion levels as there are mesh elements), we can define $I_{f,\ell} = \{i | D_i[f] = z_\ell\}$, the set of elements with a given distortion $z_\ell$. Then, the definition of ordering can be reformulated as follows.

$f \prec g$ if and only if for some $\ell$ two statements hold: (a) $z_\ell^f < z_\ell^g$ or $z_\ell^f = z_\ell^g$ and $|I_{f,\ell}| < |I_{g,\ell}|$, (b) for all $j < \ell$, $z_j^f = z_j^g$ and $|I_{f,j}| = |I_{g,j}|$, where $|\cdot|$ denotes the area: $|I_{f,\ell}| = \sum_{T \in I_{f,\ell}} A_T$.

This notion has a close relationship to *lexicographic ordering*, because if all areas $A_T$ are the same, we have $|I_{f,\ell}|$ proportional to the number of elements in $I_{f,\ell}$. If we now arrange the vector $D[\mathbf{f}]$ in the order of nonincreasing distortion values, and denote this new vector by $D[\mathbf{f}]^{lex}$, then $\mathbf{f} \prec \mathbf{g}$ if and only if $D[\mathbf{f}]^{lex}$ is lexicographically less than $D[\mathbf{g}]^{lex}$. In general, the area-weighted version we use is only slightly different.

**Computing a strict minimizer.** Let $I = \bigcup_{\ell=0}^{L'} I_{f,\ell}$ be the set of indices for which we fix an upper distortion bound $k_i$, $i \in I$, on element $i$, $L' \leq L$ is the level with the minimal distortion that we fix the distortion on (in the algorithm, the bounds $k_i$ are determined at the previous steps). We consider the set of solutions X=$M(I)$ of the problem

$$k = \min \max_{j \notin I} D_j[\mathbf{f}], \text{ subject to } D_i[\mathbf{f}] \leq k_i, i \in I \qquad (1)$$

where minimization is done with respect to $\mathbf{f}$. If $D_i[\mathbf{f}]$ are convex functions of $\mathbf{f}$, then $M(I)$ is convex. Suppose the optimal value of $\max_{j \notin I} D_j[\mathbf{f}]$ is $k$. For $M(I)$, define the set $Ess(I)$ of *essential indices*, for which $D_j[\mathbf{f}] = k$ for *all* solutions in $M(I)$.

**Proposition 1.** *$Ess(I)$ is not empty for any $I$ and convex $D_i$.*

See Appendix B for a proof. Now we can define an algorithm for computing a strict minimizer. In this form, it is clearly not practical, but it allows one to obtain a reference point for approximate solutions and will be the foundation of more efficient approximate algorithms. It also provides a foundation for proving uniqueness as described in Proposition 2.

**Algorithm 1** Strict minimization algorithm

1: $F_0 := \emptyset$, $j := 1$
2: **while** $F_{j-1} \neq \{1 \dots \mathrm{N}\}$ **do**
3: Find $Ess(F_{j-1})$ from the set of solutions $X_j$ of (1) with $I = F_{j-1}$; $k$ is the optimal value returned.
4: $F_j := F_{j-1} \cup Ess(F_{j-1})$
5: For all $i \in Ess(F_{j-1})$, $k_i := k$.
6: $j := j + 1$
7: **end while**

The first step of the algorithm solves the $L_\infty$ minimization, and identifies the set of elements with distortion at the upper bound for all solutions. The bounds for these elements are fixed. At each step, the algorithm pushes down the maximal distortion of all elements for which the bounds were not fixed, and again finds an essential subset of these. Gradually, distortion $k_i$ is determined for all elements. The end result, $X_L$ (where $L$ is the number of iterations), in fact contains all strict minimizers:

**Proposition 2.** *For convex $D_i$, any strict minimizer $\mathbf{f}^*$ has the same distortion vector $D[\mathbf{f}^*] = [k_1, \dots, k_N]$, which is obtained by Algorithm 1.*

See Appendix B for a proof. It immediately follows from the proposition that the strict minimizer is unique up to any transform of $\mathbf{f}^*$ which all $D_i$ are invariant to. While uniqueness of $D[\mathbf{f}^*]$ holds under general assumptions of $D$, an explicit description of the set $X_L$, and conditions under which it contains only one minimizer depends on $D$ and mesh connectivity.

**Relationship to the $L_p$-norms.** In the context of approximation by polynomials in one variable, when the $L_\infty$-minimizer is unique, a classic approach [Pólya 1913] is to view it as a limit of the $L_p$-norm minimizers. In the cases when the $L_\infty$ minimizer is not unique, for the setting of [Rice 1963], Descloux [1963] shows that the sequence of $L_p$ norm approximations converges to the strict uniform approximation. Fletcher et al. [1971], observing that the convergence may be extremely slow, propose an extrapolation technique to accelerate it. Thus, one can view strict minimizers in the context of surface parametrization and deformation as the limit case of $L_p$ norms, suggested in [Bommes et al. 2013] and [Gao et al. 2012] respectively. We discuss the relation in greater detail from a practical point of view in Section 6.

# 5 Algorithms

## 5.1 $L_\infty$ optimization

We develop specific algorithms for strict approximation in the context of minimizing the *field-deviation* distortion, defined in Section 4. This measure appears in a variety of deformation and parametrization contexts, both by itself and as a building block for the ARAP distortion measure discussed in Section 5.4.

The difficulty in minimizing $L_\infty$-norm, compared, e.g., to $L_2$ or any smooth functional, is that standard general methods for unconstrained optimization either converge slowly or require at least gradient information. However, $L_\infty$-norm is not smooth, although convex. We observe that in a number of cases important for geometry, in particular for optimizing $L_\infty$-norm of field-deviation distortion, $\max_{i \in I} \|J_i - R_i\|_F$, we can cast the problem as a *constrained* optimization problem, by introducing an auxiliary slack variable $k$ for the unknown $L_\infty$ bound:

$$\min k, \qquad \text{subject to}$$
$$D_i^{FD}(\mathbf{q}) = \|J_i(\mathbf{q}) - R_i\|_F \leq k, \qquad i = 1 \dots N \qquad (2)$$

where the minimization is with respect to the vector of unknowns $\mathbf{q}$. Note that the expression on the left-hand side of equalities is a 2-norm of a linear function of unknowns, and the right-hand side is a linear function. Thus, this simple transformation results in a second-order cone problem (SOCP), for which robust and efficient solvers exist. In our implementation, we use a homogeneous interior point solver (MOSEK [Andersen et al. 2003]).

While the minimal value $k$ of the $L_\infty$ norm is unique by convexity of the norm, in general many possible solutions have this optimal $L_\infty$ norm. Depending on the algorithm, different solutions $\mathbf{q}$ of (2) may be found. To define the algorithm unambiguously, we can define a unique solution as follows: (a) we solve (2) first, and use the resulting bound $k^*$ to solve a constrained $L_2$ optimization problem, $\min \sum_{i \in I} (D_i^{FD})^2 A_i$, subject to $D_i^{FD} \leq k^*$, for which the solution is unique if we fix the value at a single point (to account for invariance of $D_i^{FD}$ with respect to translations of the image).

As a building block in the strict minimization algorithm described below, we found that as far as the sets of triangles $I_i$ are concerned, the difference between simply using the solution for (2) computed by an interior point solver (MOSEK) and obtaining the constrained $L_2$ solution is insignificant (and both are far from the strict minimizer). In our experience, our $L_\infty$ algorithm is quite efficient and the solver converges with high accuracy quickly.

## 5.2 Direct Strict Minimization

We describe an *approximate* direct algorithm, following the overall structure of Algorithm 1. The key difference is the change in how the sets of elements with fixed distortion bounds $F_j$ are defined.

Identifying the set $F_j$ precisely is expensive: A common approach would be to start with an initial set $F'$ of maximal value elements for an $L_\infty$ minimizer, and attempt to minimize distortion on each element of $F'$, while keeping the rest bounded by the same bound.

First, rather than searching for precise sets $F_j$ we simply use the solution of the $L_\infty$ problem, solved as discussed in 5.1, and which, as noted, is similar to the solution of the constrained $L_2$ problem. The intuition for using a similar solution to the constrained $L_2$ problem is that the $L_2$-norm minimization is likely to lower the distortion values for all non-essential elements in the interior of the feasible domain ($L_2$-norm tends to concentrate distortion in a few spots, as this often decreases the average). While there is no guarantee that this consistently happens, Figure 3 shows that in most cases only a small number of additional triangles may be included in $F_j$ unnecessarily.

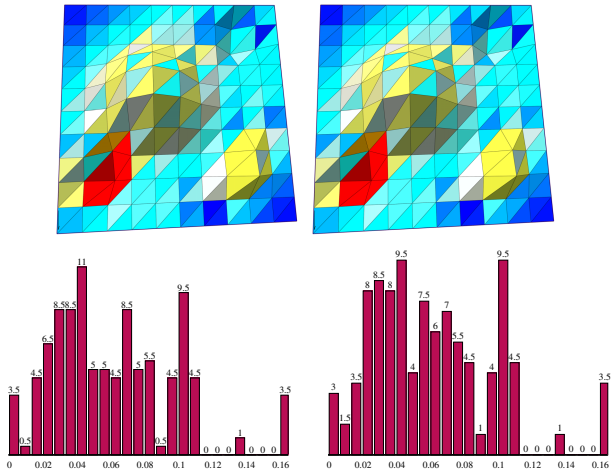


**Figure 3:** *Exact (right) vs. approximate (left) computation of sets $F_j$. The distributions of triangles by distortion (horizontal histogram axis) are nearly identical, with differences of 1-2 triangles at the most. The approximate was generated with $\Delta k = 10^{-4}$, and the exact was generated using brute force.*

The second change is to ensure that the number of steps the algorithm takes is controlled directly. Rather than fixing bounds for elements that are within solver accuracy bound from the optimum, we specify a $\Delta k$, and include in $F_j$ all elements $i$ that have distortion $D_i$ between the bound $k_j$ and $k_j - \Delta k$. This ensures that the number of steps is no more than $k_1/\Delta k$. For example, typically for high-quality fields $R_i$, $k_1$ is 1 or below, and using $10^{-3}$, we can restrict the number of steps needed to 1000. We observe that typically the number of steps is substantially lower, as there are often gaps in the distortion value distribution. Further more, we set a lower distortion bound $k_{min}$, where the distortion becomes negligible, and stop the iterative process prematurely. Nevertheless, the algorithm in this form remains quite costly but quite accurate, yielding a close approximation to the true strict minimizer.

**Algorithm 2** Direct approximate minimization

$F_0 := \emptyset$, $j := 1$, $k := \infty$, $k_{min} := const$.
**while** $F_{j-1} \neq \{1 \dots \mathrm{N}\}$ and $k > k_{min}$ **do**
  Solve (1) with $I = F_{j-1}$ to obtain the optimal value $k$.
  $E = \{i | k - \Delta k \leq D_i^f \leq k\}$
  $F_j := F_{j-1} \cup E$
  For all $i \in E$, $k_i := k$.
  $j := j + 1$
**end while**

### 5.3 Relaxation-Based Strict Minimization

A significantly faster algorithm is based on a more direct application of the definition of the strict minimizer. Recall that the strict minimizer has minimal vector of per-element distortion values with respect to the relation $\preceq$ defined in Section 4. Hence, a natural approach would be to find a starting $L_\infty$ minimizer and then iteratively improve it, always transitioning from a solution $\mathbf{q}$ to a new solution $\mathbf{q}'$ such that $\mathbf{q}' \prec \mathbf{q}$. The simplest algorithm of this type is coordinate descent. We keep all components of $\mathbf{q}$ fixed, and change only one.

We use *1-ring* $L_\infty$ minimization as an elementary operation to modify $\mathbf{q}$. For a 1-ring $N_1(v)$ centered at a vertex $v$, we solve a small $L_\infty$ problem with the value at $v$ as the unknown, and values at all other vertices fixed. In this respect, the process resembles many similar algorithms ranging from Gauss-Seidel iteration to the classical flip algorithm for Delaunay triangulation.

With this process, only the distortion on the triangles in the 1-ring we optimize can change. Generically, the minimizer for the 1-ring L-infinity problem is unique. Either the 1-ring is already at the minimum – then $\mathbf{q}$ does not change – or it decreases the maximal distortion $k$ in the 1-ring, which ensures that the new $\mathbf{q}'$ is less distorting. It will decrease the area $H_{f,k}$, and any possible increase in distortion will be at values less than $k$.

We refer to this simple algorithm as relaxation-based strict minimization, as it gradually progresses towards the strict minimizer by local improvements. We note that the principle underlying the flip algorithm for Delaunay is the same. The Delaunay triangulation minimizes the maximal angle (i.e., solves a discrete $L_\infty$ problem), and if the distortion vector is defined as the vector of all triangle angles, then a single flip results in a lower distortion vector with respect to the $\prec$ relation that we have defined.

**Algorithm 3** Relaxation-based approximate minimization

Solve $L_\infty$ optimization problem to obtain solution $\mathbf{q}$
**for** $m = 1 \dots m_{max}$ **do**
  **for** $v \in V$ **do**
    Solve 1-ring $L_\infty$ problem on $N_1(v)$, for $\mathbf{q}_v^*$.
    Replace $\mathbf{q}_v$ with $\mathbf{q}_v^*$.
  **end for**
**end for**

The algorithm is easily parallelized in a red-black relaxation style (see performance discussion in Section 4 in [Levi and Gotsman 2015]). A stopping condition can be a threshold on the maximum change in an iteration over all the 1-rings. In our experiments, we found that setting $m_{max} = 50$ is more robust, and it is sufficient in most cases for reaching a close to steady state for high distortion values. This is attributed to the behavior of the algorithm that is similar to many relaxation algorithms. It quickly reduces high distortions, but proceeds slowly after that, gradually decreasing lower distortions in a broader area, which we care less about. Further benefits of the algorithm include: (i) it can be used to improve an initial solution, which was generated with $L_p$ or IRLS (discussed in Section 6); (ii) the algorithm can be used in a multiresolution scheme; (iii) it can be modified to favor solutions with fewer flips or avoid adding flips to an initial solution altogether.

**Solving 1-ring problems in 2D.** 1-ring optimization problems with three (2D) or four (3D) variables can be solved quickly using a variety of standard algorithms. In 2D, for field-deviation distortion or distortion measures with similar properties, the 1-ring optimization problems can be solved particularly efficiently, due to their special properties. We briefly describe a specialized function that we use, with detailed analysis in the supplementary document.

Suppose $p_1, p_2, p_3$ are 3 vertices of a triangle $T$ (in a local 2D coordinate system), $q_1$ and $q_2$ are fixed parametric-plane images of $p_1$ and $p_2$, and $q = [u, v]$ is the vertex that is allowed to move.

**Proposition 3.** *The set of positions of point $q$ for which distortion is equal to a given $k$ is either empty, or a circle; the center of the circle does not depend on $k$.*

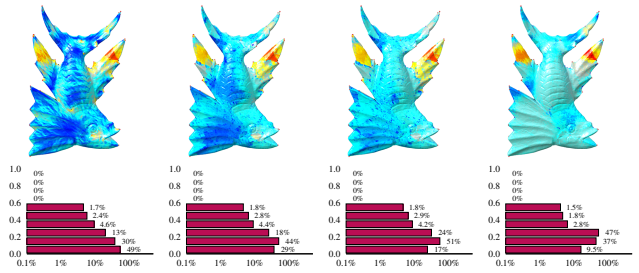


**Figure 4:** *Progress of relaxation-based strict minimizer: iterations 0 ($L_\infty$ minimizer), 40, 140, and the result of the direct strict minimizer. Note that solutions do not differ significantly at high distortion levels, but differ more at lower levels (all histograms are log-scale, with distortion amount vertical). Here we chose an example where there is a clear difference between 40 and 140 iterations; usually 50 iterations are enough.*

The geometric notion of distortion circles allows to define the space of feasible solutions of the constraints above in the 3D space $(q, k) = (u, v, k)$, This is a set of circular cones, with vertical axes (along $k$), with apexes at different heights $k_i$ (the height of an apex is the minimal possible distortion for a triangle in the one ring). The feasible domain is the intersection of feasible cones for all triangles in the ring $T_i$, $i = N$. This is a convex 3D domain with piecewise smooth boundary, and the solution of the optimization problem is the lowest point of this domain. As the boundary is made of pieces of circular cones, the following proposition holds:

**Proposition 4.** *The solution $q^*, k^*$ of the 1-ring $L_\infty$ problem is at a vertex of the feasible domain (intersection of 3 distortion cones), at an edge point (lowest point of the intersection curve of two cones), or at a cone apex.*

The intersection of two or three circular cones with parallel axes can be found by solving quadratic equations for the intersection points. Similarly, finding the lowest point of an intersection curve amounts to solving a quadratic equation. A brute force check of all possible minima locations requires solving 41 quadratic equations for a vertex of valence 6. While the cost of such brute force algorithm is cubic in the size of the neighborhood, as the fraction of high-valence vertices is typically small, actual performance is close to the performance one expects for the average case.

### 5.4 ARAP Optimization

So far, we have considered convex distortion measures, focusing on field-deviation distortion. Nevertheless, the algorithms of the previous section are applicable to a broader range of functionals, e.g.:

$$D^{ARAP} = \min_{R \in SO(n)} \|J - R\|_F \tag{3}$$

For 2D maps, this measure can be expressed as $\sqrt{(\sigma_1 - 1)^2 + (\sigma_2 - 1)^2}$ in terms of singular values of the Jacobian $\sigma_1$ and $\sigma_2$, and similar expressions hold in 3D. The algorithm described in the previous section can be easily integrated with a commonly used local-global ARAP iteration (e.g., [Liu et al. 2008]. The local step in the ARAP algorithm updates the frames using

$$R_i := \operatorname*{argmin}_{R \in SO(n)} \|J_i(\mathbf{f}) - R\|_F, \tag{4}$$

where the map $\mathbf{f}$ is kept fixed, while the global step minimizes field-deviation distortion keeping $R_i$ fixed.

We observe that $L_\infty$ optimization of the field-deviation distortion is a natural match for local-global iteration: The local frame update *does not increase* the distortion on any element, so a solution satisfying constraints with some bounds $k_i$, still satisfies these bounds after local update. While it is not guaranteed to converge to the global minimum of the nonconvex energy, like similar local-global algorithms, it is guaranteed to go to less distorted mappings at every step. Similar observation holds for the 1-ring $L_\infty$ minimization.

Thus to obtain an accurate algorithm for approximating the strict minimizer of ARAP distortion, we replace a single call to $L_\infty$ optimization in Algorithm 2 by a local-global iteration. Similarly, for the fast relaxation-based algorithm, we add after the inner loop in Algorithm 3 a local iteration. We can guarantee that for each iteration $\mathbf{q}' \preceq \mathbf{q}$, because the local step decreases (or at least does not increase) each component of the distortion vector.

**ARAP shape interpolation.** Using the method of [Levi and Gotsman 2015], the ARAP distortion measure can be easily extended to the problem of shape interpolation. Suppose we are given two meshes $M_1$ and $M_2$ of the same connectivity, with a correspondence between the vertices of the two, defining a piecewise linear map $f$ with differential $J_i$ on the element $T_i$. Let $R_i^1 Y_i^1$ be a polar decomposition of $J_i$. [Levi and Gotsman 2015] defines the symmetric part of the intermediate map $f_t$, $Y_i^t = (1-t)Y_i^0 + tY_i^1$, for $0 \le t \le 1$, where, $Y_i^0 = I$, is the symmetric part of the Jacobian of the initial map (identity). The intermediate shapes at time $t$ are obtained by optimizing the ARAP-like local distortion measure

$$\min_{R_i \in SO(3)} \|J_i^t - R_i Y_i\|_F$$

where $R$ is a rotation. Our algorithms can be applied directly in this setting. More generally, approximate strict minimizers can be computed for any functional that can be optimized by a local-global algorithm by a similar modification of our algorithms.

## 6 Evaluation

For evaluation, we use primarily global parametrization (see Appendix A) examples. The fields for these examples were obtained using the algorithm of [Bommes et al. 2009]. In all cases, we use $k_{min} = 0.1$, $\Delta k = 10^{-3}$ for the direct algorithm, and 50 iterations of the iterative algorithm. In the images, the distortion is color-coded on the facets of the source mesh: dark blue corresponds to distortion 0 and red to distortion 0.6. All histograms are log-scale, with horizontal axis indicating percentage of triangles at a given distortion, and vertical the distortion magnitude.

**Controlling distortion of quadrangulations.** One of the motivations for this work is the difficulty with controlling bijectivity and more generally distortion of global parametrizations and quad remeshing. Minimizing average distortion results in high distortion concentrations which typically cannot be eliminated by using a global constraint on distortion. Figs. 6-7 demonstrates that a significant improvement in quad quality is achieved by our method. While we show examples with relatively moderate (e.g., by finite element standards) quad distortion, similar effects are observed for higher distortion levels. In all cases we observe a large improvement both of aspect ratio and size uniformity. We note that our method, in pure form, does *not* guarantee that the parametrization is bijective unless the resulting maximal field-deviation distortion is less than 1. However, it is easy to add other types of constraints [Lipman 2012; Bommes et al. 2013] that ensure that the result is bijective (if a feasible solution is found). The stability of the method is illustrated in Fig. 5. A different triangulation barely influenced the histogram, and the added noise made a difference relative to its strength.

**$L_\infty$-minimizer vs. strict minimizers.** Next, we compare the effects of our algorithms; see Figs 8-9. We show the effects on the distortion histograms, which can be viewed as averaged representations of distortion vectors $D^f$, as well as some examples of local effects on the maps. Observe consistent reduction in the numbers of high (but not maximal!) distortion triangles, at the expense of

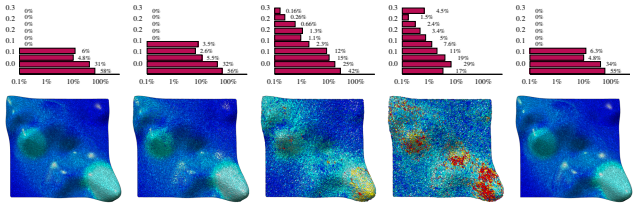


**Figure 5:** *Stability. From left to right: source model, noise $\alpha = 0.1$, noise $\alpha = 0.3$, noise $\alpha = 0.5$, remeshed (different triangulation). The noise was added as a displacement vector in the normal direction (or its opposite) of length up to the average edge length scaled by $\alpha$.*

| model | #faces | $L_\infty$(s) | direct,# | direct (s) | relax. (s) |
|---|---|---|---|---|---|
| cow | 8k | 1.3 | 28 | 45 | 1.2 |
| dragon | 104k | 18 | 59 | 1810 | 15 |
| elephant | 50k | 9 | 34 | 558 | 7 |
| hand | 3k | .3 | 58 | 6 | .3 |
| homer | 10k | 1 | 43 | 67 | 1.3 |
| horse | 102k | 17 | 49 | 1480 | 13 |
| igea | 100k | 19 | 99 | 3208 | 13 |
| julius | 85k | 19 | 22 | 657 | 11 |
| lion | 55k | 8 | 24 | 363 | 8 |
| piggy | 10k | 1.5 | 115 | 125 | 1.8 |
| neptune | 105k | 16 | 69 | 2003 | 15 |
| fish | 35k | 6 | 54 | 354 | 5 |

**Table 1:** *We ran the experiments on a laptop with Intel Core i7-2720QM CPU 2.2Ghz. For the relaxation-based algorithm, we ran 50 iterations.*

an increase in distortion at low levels. This increase in distortion at low levels, while nearly imperceptible and typically not reducing, e.g., finite element accuracy, at the same time leads to a significant increase in $L_2$-norm. We also observe that while the histogram behavior for the relaxation-based algorithm for low distortion levels can be quite different, it achieves similar effects for high distortion level.

**ARAP parametrization, deformation, and shape interpolation.** The results for ARAP global parametrization are shown in Figure 17. As expected, a reduction in distortion is achieved, and the overall behavior is similar to the field-deviation case. A simple example for shape deformation, based on ARAP distortion measure, is shown in Figures 10-11; observe that essentially uniform distortion is obtained. Similar behavior is observed for shape interpolation (Figure 12).

**Performance.** The information on performance of our algorithms is presented in Table 1, which includes most of the shapes used in this paper. The column *direct,#* shows the number of steps in the direct algorithm. Observe that the time required by the relaxation-based algorithm (not counting initialization) is about the same as a single $L_\infty$ solve.

Our $L_\infty$ optimization method yields significant performance advantages compared to [Aigerman and Lipman 2013], discussed in more detail in the next section: we can obtain an approximation of a strict minimizer with a fraction of the cost of a single projection iteration.

**Alternative algorithms for strict minimization.** We consider two potential alternatives to the algorithm that we propose. One is Polya's algorithm which we have already discussed in Section 4. It requires solving a sequence of $L_p$ problems to determine the strict minimizer as a limit. One could consider simply solving a single $L_p$ problem for a large $p$. However, we observe (Figure 13) that the $L_p$ minimizers approach the strict minimizer first, but then start moving away from the strict minimizer. While experimentally we see a

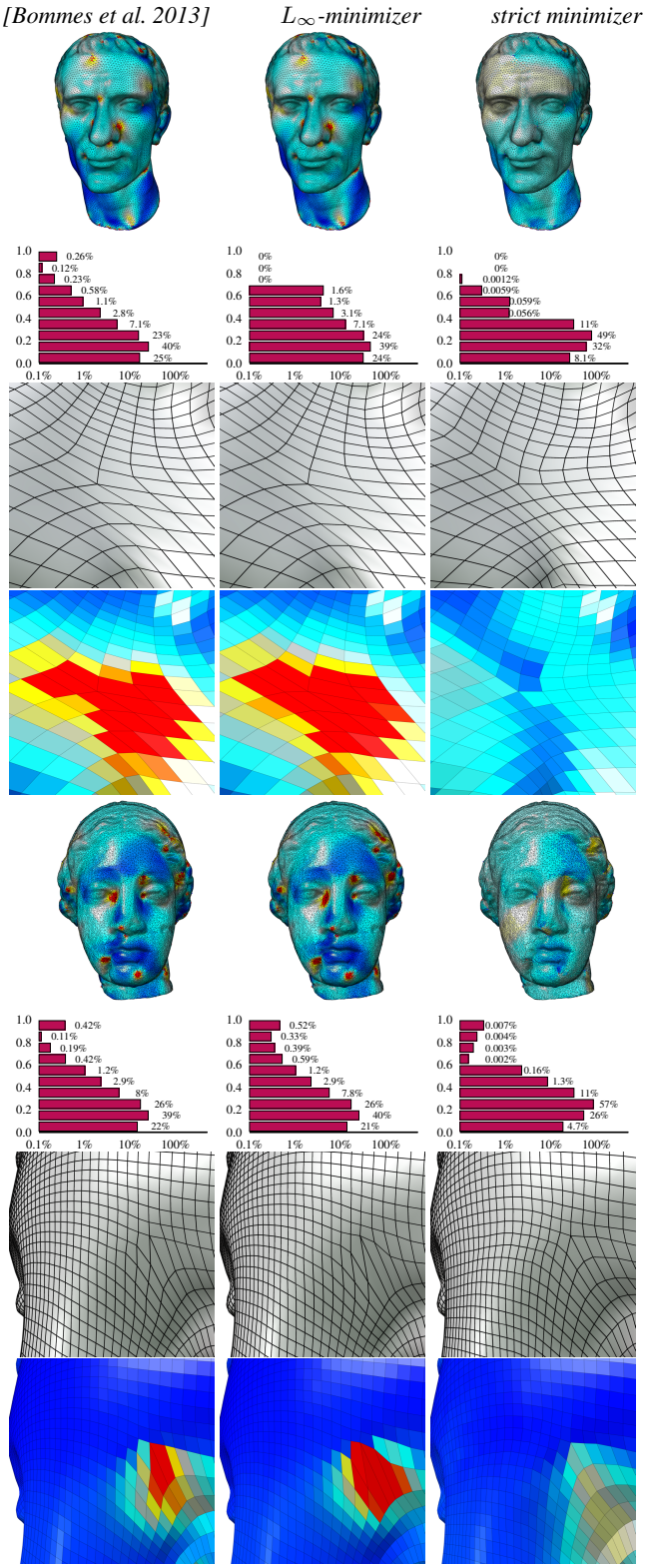



**Figure 6:** *Comparison of strict minimization algorithms to the method of [Bommes et al. 2013] and $L_\infty$ minimizer. (In the first row, the increased maximum distortion in the strict minimizer is due to accumulated numerical errors right on the border between the histogram bins.)*

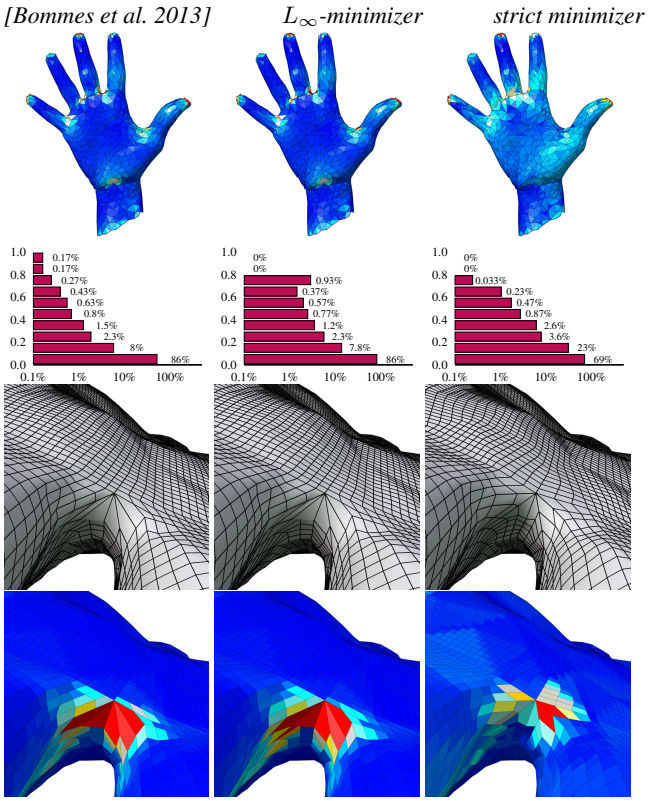



**Figure 7:** *An additional example of parametrization rounding*

reasonably close approximation to strict minimizer for $p = 16$ or 32, the maximal distortion for these values is always higher than the expected limit value. For higher values of $p$, the maximal distortion gets closer, but the number of triangles with distortion close to the highest increases, and the $L_p$ solution deviates from the strict minimizer. Even a single stage of $L_p$ optimization is 5-6 times slower than the relaxation-based algorithm, and at least several minimizations of $L_p$ are required if we want to identify for which value of $L_p$ the result is optimal, in the sense of the distortion ordering we have defined.

Another possible option is using Iterative Reweighted Least Squares (IRLS). A version of IRLS for $L_\infty$ optimization was proposed by Lawson [Rice and Usow 1968], however, it is not known if it converges to a strict minimizer or not. As [Rice and Usow 1968] points out, its rate of convergence depends on the ratio of the worst distortion to second worst, which is likely to be small in the problems we consider. We observe that it does yield some improvement but remains far from the strict minimizer.

Finally, we observe that the stiffening algorithm of [Bommes et al. 2009] can be viewed as a type of IRLS algorithm, but with weights determined by a different formula. The results of using IRLS appear to be better, but in both cases, a more significant reduction of distortion can be obtained using our approximations of strict minimizers.

## 7 Other Distortion Measures

So far, we considered a strict minimizer for our ARAP-$L_\infty$ method. But the strict minimizer concept can be easily extended to other algorithms and distortion measures. We consider two different distortion measures in 2D, quasiconformal distortion [Lipman 2012] and an alternative measure of deviation from isometry [Aigerman and Lipman 2013]. To evaluate the behavior of strict minimizers with these measures, we use a binary-search (bisection) $L_\infty$-optimization in the direct strict optimization solver of Section 5. It is possible to improve the results using the local-global framework, but due to the bisection it would take too long.

**Quasiconformal distortion** (or dilatation) is a particularly interesting test case, because of the difference in the behavior of $L^\infty$-minimizers in the smooth and discrete cases. This measure is defined by the deviation from conformality of an affine map $f$, $D^{QC} = \sigma_1/\sigma_2$, where $\sigma_1 \geq \sigma_2$ are the largest and smallest singular values of the Jacobian of $f$. In the smooth case, the solution of the $L_\infty$ optimization problem is, in fact, unique and is a *Teichmüller map* [Weber et al. 2012], characterized by $D^{QC} = const = k$ everywhere, excluding isolated points where the map is not smooth. No analogous statement is known in the discrete case. There are many $L_\infty$ minimizers, and an optimization does not produce maps that are anywhere close to uniform distortion. There is a simple way to evaluate the quality of a discrete solution, by considering how close to constant $D^{QC}$ is. The discrete strict minimizer results in (Figure 15) are much closer approximation to the smooth Teichmüller map vs. the $L_\infty$ minimizer. Although the results of [Weber et al. 2012], which explicitly tries to make $D^{QC}$ constant, are also close to constant distortion, the algorithm proposed there does not guarantee bijectivity and, as can be observed, has a small number of triangles deviating from average. In contrast, a strict minimizer has much lower maximal distortion.

**ASAP.** As-similar-as-possible distortion measure is a different way to measure conformal distortion. Similar to (3) we define

$$D^{ASAP} = \min_{R\in SO(n), s\in\mathbb{R}^+} \|J - sR\|_F \quad . \tag{5}$$

We modify the local step in the framework suggested in Section 5.4 to update the frames to the closest similarity matrix instead of a rotation

$$s_i R_i := \operatorname*{argmin}_{R\in SO(n), s\in\mathbb{R}^+} \|J_i(\mathbf{f}) - sR\|_F \quad . \tag{6}$$

For 2D maps, the ASAP distortion measure in (5) can be expressed as $\sigma_1 - \sigma_2$; expressions for 3D maps are similar. As minimizing the ASAP distortion is another way to minimize the conformal distortion, the strict minimizer approximates a Teichmüller map (Figure 15).

**Bounded singular values.** A different measure of deviation from isometry than ARAP is used in [Aigerman and Lipman 2013], $D^{SVM} = \max(\sigma_1, 1/\sigma_2)$. Since the constraint, $D^{SVM} \leq k$, is not convex, a projection of a given map to the bounded space is approximated. The strict minimizer is demonstrated in Figure 16. While there is, as expected, some decrease in high distortion, it is modest compared to what is observed for $D^{FD}$, $D^{ARAP}$, and $D^{QC}$.

**Singular-value bound vs. $L_\infty$-ARAP optimization.** Our approach to minimizing the $L_\infty$ norm of the ARAP distortion measure ($L_\infty$-ARAP) provides a useful alternative to methods based on optimizing bounds on singular values as in [Aigerman and Lipman 2013]. These methods yield different types of bounded distortion maps. We compare these approaches in more detail to clarify relative benefits of each.

- Both methods yield results with a reasonable trade-off between conformal and area distortion, and work in any dimension.
- While [Aigerman and Lipman 2013] can operate inside the local-global framework (Section 5.4), $L_\infty$-ARAP optimization in the local-global setting ensures that the energy is decreased at every step.
- The $L_\infty$-ARAP is quite efficient, as it requires solving a single SOCP in each global step, finding the tightest bound for

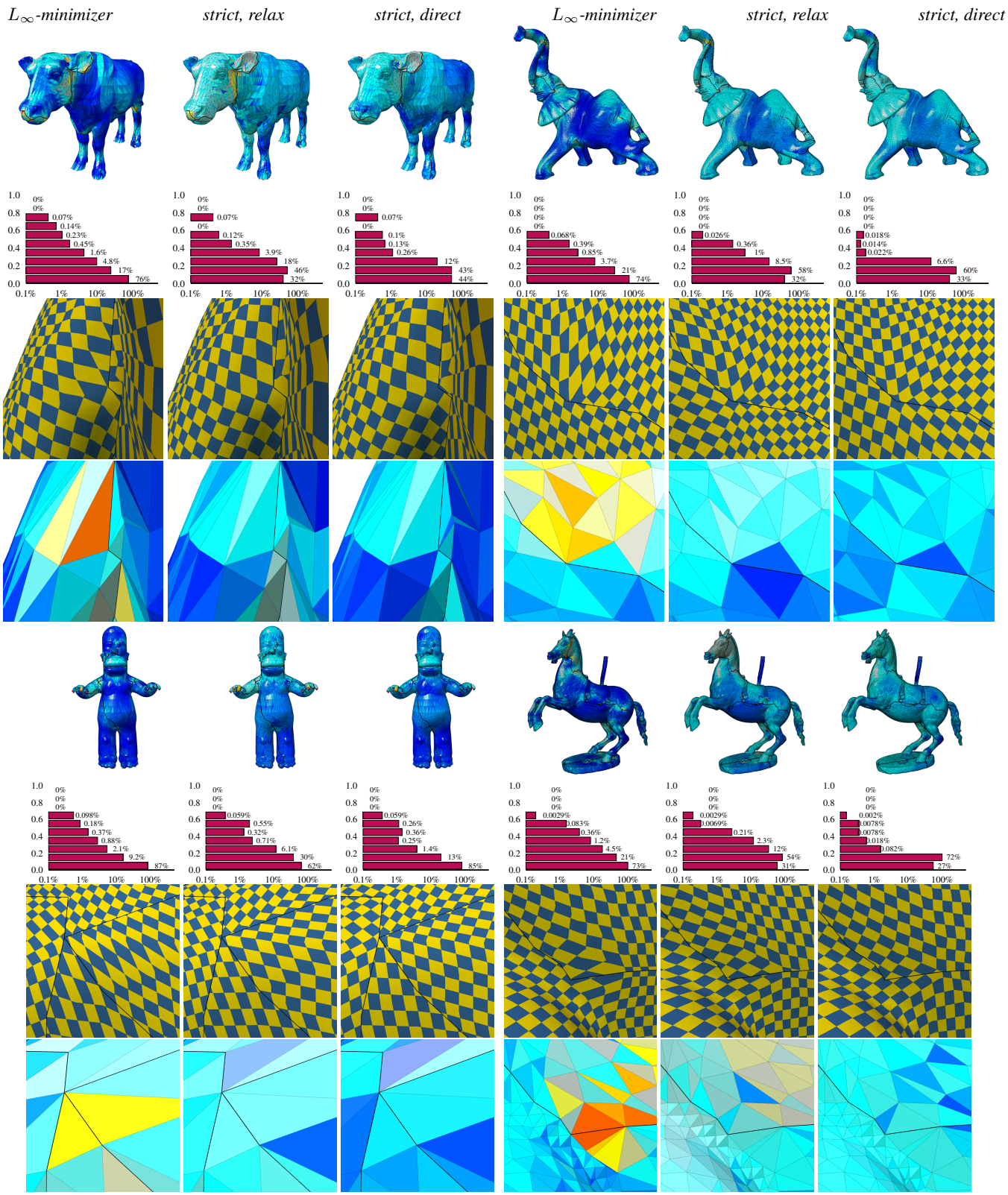



**Figure 8:** *$L_\infty$ optimization, relaxation-based, and direct algorithms for strict minimization for global mesh parametrization. Please note that no singularity position rounding is done, so the displayed textures have a mismatch along the seam, which was used to cut the surface to a disk.*

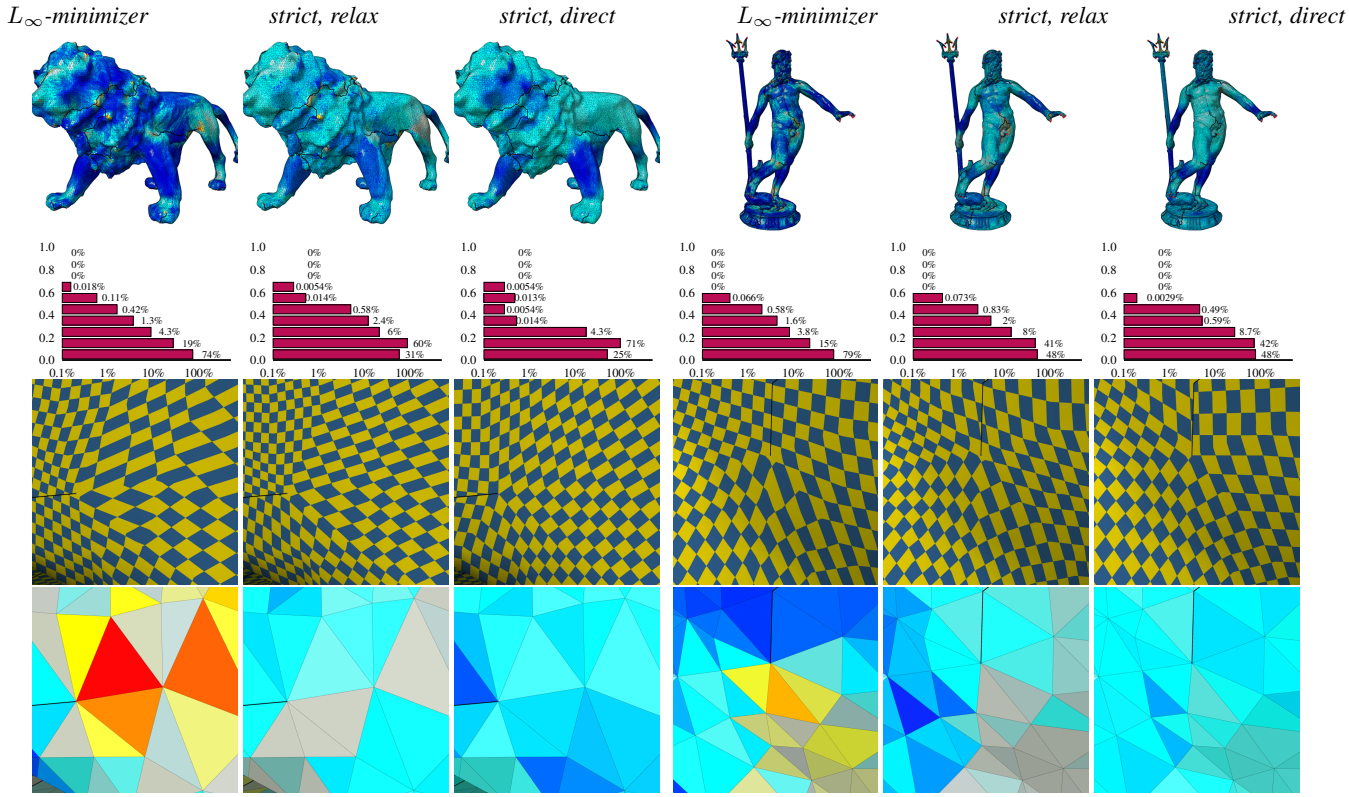



**Figure 9:** *Additional examples comparing $L_\infty$ optimization, relaxation-based, and direct algorithms examples.*

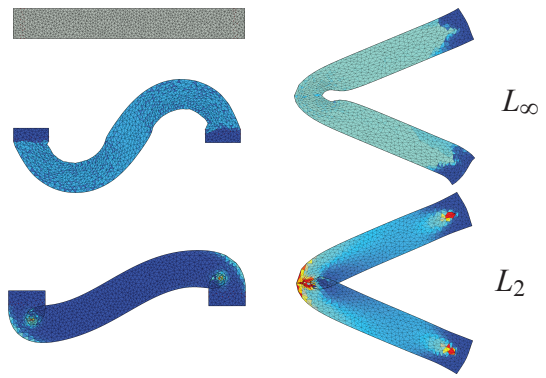



**Figure 10:** *Two ARAP deformation examples; the result is colored by distortion. Note that in these cases the distortion is essentially constant for our algorithm. The $L_2$ results contain foldovers.*

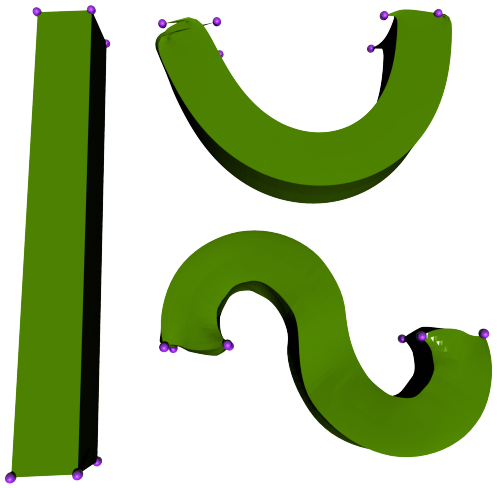


**Figure 11:** *Tetrahedral bar deformation. (Left) source, (Top) $L_2$ concentrates the error near the positional constraints, (Bottom) $L_\infty$ keeps the error bounded.*

the step, without projections and bisection iterations needed by [Aigerman and Lipman 2013].

- $L_\infty$-ARAP problem is always feasible. For example, the initialization step of the local-global framework is usually arbitrary, and the frames (rotations) are set to identity. Thus, it is common in deformation applications to go through unavoidable configurations with inverted elements, which are not in the feasible space of problems solved in [Aigerman and Lipman 2013]. For example, the two bar deformations in Fig. 10 cannot be obtained with a method that does not allow intermediate configurations with inverted elements.
- [Aigerman and Lipman 2013] guarantees, when a feasible solution is found, that it is bijective. The $L_\infty$-ARAP minimizer may not be bijective even if a bijective solution exists (bijectivity is guaranteed only if the optimal norm is lower than 1).

## 8 Conclusions and Future Work

Strict minimizers introduced in this paper provide a natural definition of uniquely defined maps of least distortion, in cases when the space of maps with the best possible distortion bound is large. While the algorithms closely following the definition are expensive, we proposed an alternative that is easy to implement, efficient, and

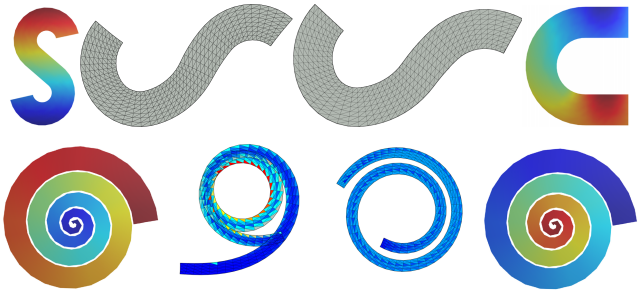


**Figure 12:** *Shape interpolation. (First row) middle shapes for* $t = 0.5$*, [Chen et al. 2013] and* $L_\infty$*, (Second row)* $t = 0.75$*,* $L_2$ *and* $L_\infty$*.*

gives a reasonable approximation to the exact solution.

Many improvements and additional applications to 2D and 3D deformation and shape interpolation are possible. We can expect that a simplex-type algorithm combined with our algorithm for one-ring optimization would result in large speedups, reducing the cost to just several quadratic equation solves per one ring, making its cost substantially less than a single SOCP solve. A multiresolution relaxation-based algorithm can be constructed both reducing the number of iterations and potentially improving quality.

While strict minimizers in pure form yield a useful class of mappings, in practice it may be necessary to combine these with other quality functionals or impose additional constraints (e.g., use them locally to improve a given map).

From a theoretical point of view, nothing is known about the limit behavior of strict minimizers under mesh refinement. In fact, a number of examples show that some of the properties cannot be transferred exactly. For this reason, excluding special cases such as Teichmüller maps, nothing is known at this point about the smoothness of the limit solutions.

## Acknowledgements

We would like to thank the people who gave us useful comments, especially Julian Panetta who assisted us with the proofs. This research was supported by NSF awards IIS-1320635 and IIS-1247240.

## References

ABDELMALEK, N. N. 1977. Computing the strict Chebyshev solution of overdetermined linear equations. *Mathematics of Computation 31*, 140, 974–983.

AIGERMAN, N., AND LIPMAN, Y. 2013. Injective and bounded distortion mappings in 3D. *TOG 32*, 4, 106.

ALEXA, M., COHEN-OR, D., AND LEVIN, D. 2000. As-rigid-as-possible shape interpolation. In *Computer graphics and interactive techniques*, 157–164.

ANDERSEN, E., ROOS, C., AND TERLAKY, T. 2003. On implementing a primal-dual interior-point method for conic quadratic optimization. *Mathematical Programming 95*, 2, 249–277.

BEHRINGER, F. 1981. A simplex-based algorithm for the lexicographically extended linear maxmin problem. *Euro. J. Operational Research 7*, 3, 274–283.

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

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

BOMMES, D., LÉVY, B., PIETRONI, N., PUPPO, E., SILVA, C., TARINI, M., AND ZORIN, D. 2012. State of the art in quad meshing. In *Eurographics STARS*.

BOMMES, D., CAMPEN, M., EBKE, H.-C., ALLIEZ, P., KOBBELT, L., ET AL. 2013. Integer-grid maps for reliable quad meshing. *ACM Trans. Graph. 32*, 4.

CHAO, I., PINKALL, U., SANAN, P., AND SCHRÖDER, P. 2010. A simple geometric model for elastic deformations. *ACM Trans. Graph. 29*, 4 (July), 38:1–38:6.

CHEN, R., WEBER, O., KEREN, D., AND BEN-CHEN, M. 2013. Planar shape interpolation with bounded distortion. *ACM Trans. Graph. 32*, 4 (July), 108:1–108:12.

DESBRUN, M., MEYER, M., AND ALLIEZ, P. 2002. Meshes and parameterization: Intrinsic parameterizations of surface meshes. *Computer Graphics Forum 21*, 3, 209.

DESCLOUX, J. 1963. Approximations in $L^p$ and Chebyshev approximations. *Journal of the Society for Industrial & Applied Mathematics 11*, 4, 1017–1026.

FLETCHER, R., GRANT, J., AND HEBDEN, M. 1971. The calculation of linear best $L_p$ approximations. *The Computer Journal 14*, 3, 276–279.

GAO, L., ZHANG, G., AND LAI, Y. 2012. $l_p$ shape deformation. *Science China Information Sciences 55*, 5, 983–993.

HORMANN, K., AND GREINER, G. 1999. MIPS: An efficient global parameterization method. *Curve and Surface Design: Saint-Malo 2000*, 153–162.

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* (April), 412–438.

LAWSON, C. L. 1972. Transforming triangulations. *Discrete mathematics 3*, 4, 365–372.

LEVI, Z., AND GOTSMAN, C. 2015. Smooth rotation enhanced as-rigid-as-possible mesh animation. *TVCG*.

LÉVY, B., PETITJEAN, S., RAY, N., AND MAILLOT, J. 2002. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph. 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. J. 2008. A Local/Global approach to mesh parameterization. *Computer Graphics Forum 27*, 5 (July), 1495–1504.

LUSS, H., AND SMITH, D. R. 1986. Resource allocation among competing activities: A lexicographic minimax approach. *Operations Research Letters 5*, 5, 227–231.

MARANO, M. 1990. Strict approximation on closed convex sets. *Approximation Theory and its Applications 6*, 1, 99–109.

MARCHI, E., AND OVIEDO, J. A. 1992. Lexicographic optimality in the multiple objective linear programming: The nucleolar solution. *Euro. J. Operational Research 57*, 3, 355–359.

MÉMOLI, F., SAPIRO, G., AND THOMPSON, P. 2006. Geometric surface and brain warping via geodesic minimizing lipschitz extensions. In *MFCA*, 58–67.

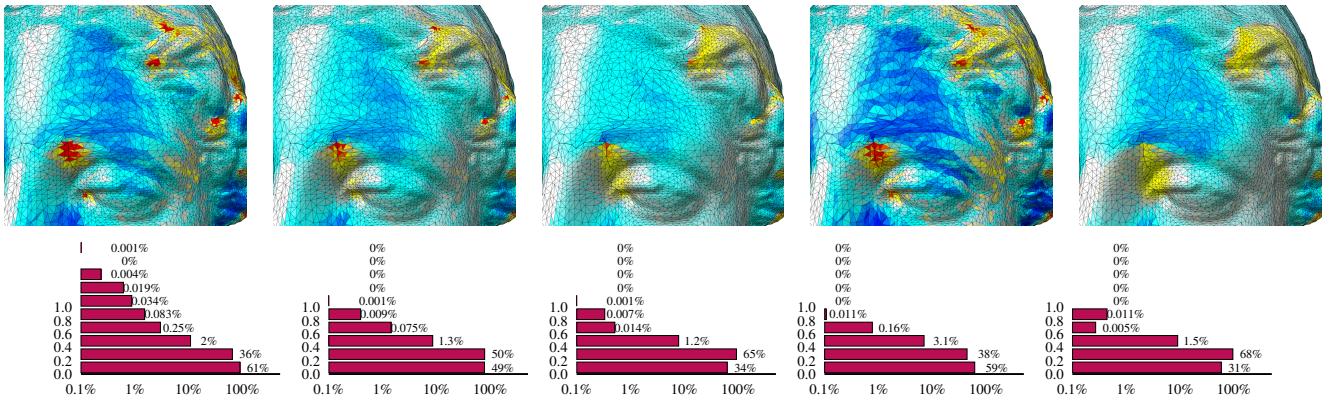



**Figure 13:** *From left to right: Distortion distribution for minimizing the $L_p$-energy, for p =4, 8, 16, 128, and strict minimizer.*

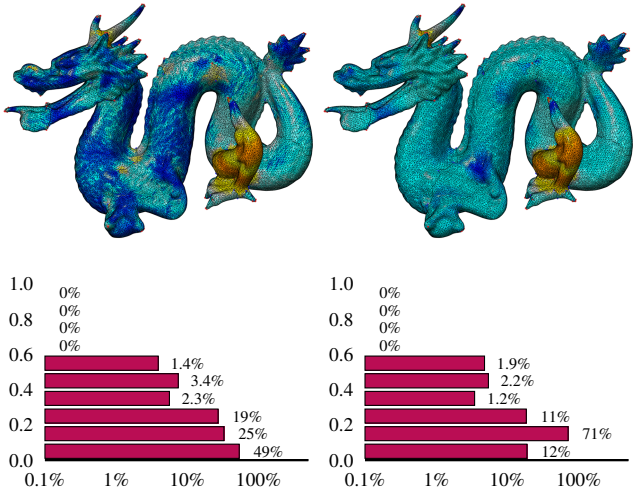



**Figure 14:** *Iterative Reweighted Least Squares after 100 iterations compared to the strict minimizer.*


MYLES, A., AND ZORIN, D. 2012. Global parametrization by incremental flattening. *TOG 31*, 4, 109.

MYLES, A., AND ZORIN, D. 2013. Controlled-distortion constrained global parametrization. *TOG 32*, 4, 105.

OGRYCZAK, W., AND SLIWINSKI, T. 2007. Lexicographic max-min optimization for efficient and fair bandwidth allocation. In *International Network Optimization Conference (INOC)*.

PÓLYA, G. 1913. Sur un algorithme toujours convergent pour obtenir les polynômes de meilleure approximation de tchebycheff pour une function continue quelconque. *CR Acad. Sci. París 157*, 840–843.

RAY, N., LI, W., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *TOG 25*, 4, 1460–1485.

RICE, J. R., AND USOW, K. H. 1968. The lawson algorithm and extensions. *Mathematics of Computation 22*, 101, 118–127.

RICE, J. R. 1962. Tchebycheff approximation in a compact metric space. *Bull. American Mathematical Society 68*, 4, 405–410.

RICE, J. R. 1963. Tchebycheff approximation in several variables. *Tran. American Mathematical Society 109*, 3, 444–466.

SCHÜLLER, C., KAVAN, L., PANOZZO, D., AND SORKINE-HORNUNG, O. 2013. Locally injective mappings. *CGF 32*, 5, 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.

SOLOMON, J., BEN-CHEN, M., BUTSCHER, A., AND GUIBAS, L. 2011. As-killing-as-possible vector fields for planar deformation. In *CGF*, vol. 30, 1543–1552.

SORKINE, O., AND ALEXA, M. 2007. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, vol. 4.

SORKINE, O., COHEN-OR, D., GOLDENTHAL, R., AND LISCHINSKI, D. 2002. Bounded-distortion piecewise mesh parameterization. In *Proc. IEEE Visualization '02*, 355–362.

SPRINGBORN, B., SCHRÖDER, P., AND PINKALL, U. 2008. Conformal equivalence of triangle meshes. *TOG 27*, 77:1–77:11.

WEBER, O., MYLES, A., AND ZORIN, D. 2012. Computing extremal quasiconformal maps. In *CGF*, vol. 31, 1679–1689.


## A Algorithmic Details for Global Parametrization

For global parametrization, the overall structure of the algorithms remains the same, but there are several differences. We largely use the formulation of [Bommes et al. 2009]. Recall that $\mathbf{f}$ consists of values $f_{im}$ assigned to corners of each element $T_i$ (in this case, parametric coordinates), where $i$ is a vertex and $m$ is a triangle index. For every interior edge of the mesh, an integer *matching* $p_{ij}$ is defined, which determines how parametric coordinates change across the edge $e_{ij}$. Typically, matchings are inferred from a field [Bommes et al. 2012]. Specifically, let $v_i$ be a vertex of an edge $e_{ij}$ shared by triangles $T_\ell$ and $T_m$. Let $f_{i\ell}$ and $f_{im}$ be parametric positions of $v_i$ at corners of $T_\ell$ and $T_m$. For each edge, these are related by the constraint

$$f_{i\ell} = R_{ij} f_{im} + t_{ij} \tag{7}$$

where $R_{ij}$ is a rotation by $\pi p_{ij}/2$, and $t_{ij}$ are auxiliary variables, referred to as *translations*. The system $C$ consists of these constraints for all edges (in practice only edges with nontrivial $R_{ij}$ yield explicit constraints). The matrix $C$ is assembled and all convex problems are solved with these linear constraints added. $t_{ij}$ are regarded as free variables. To use a global parametrization for quadrangulation, we need an integer grid layout, i.e., $t_{ij}$ need to have integer components. For the quadrangulation examples shown in the paper, these values of $t_{ij}$ are computed using the solver described in [Bommes et al. 2009], and then fixed for strict minimization.

Finally, we note that for 1-ring optimization, the center of the ring $v_i$ might have multiple positions $f_{im}$. Then, using (7) with fixed $t_{ij}$, we can express $J_m$ for all triangles $T_m$ as functions of $f_{i1}$,

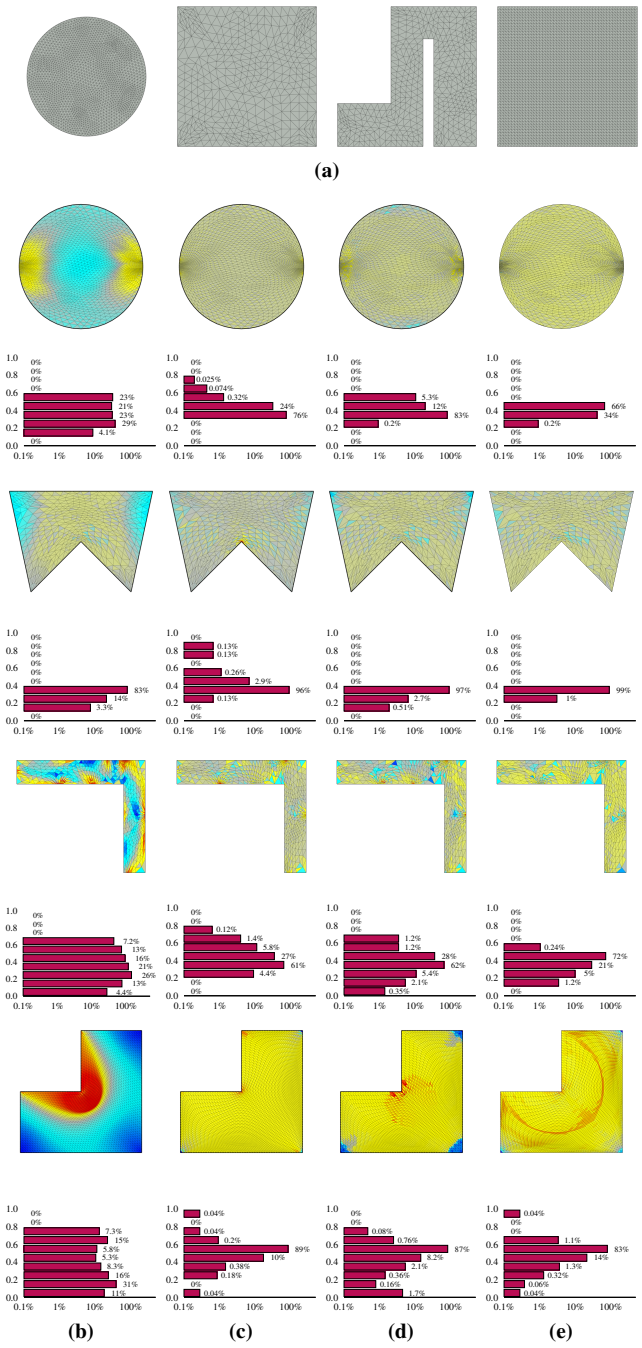



**Figure 15:** *Computing discrete approximations to Teichmüller maps. (a) Source domains of the three maps, (b) [Lipman 2012], (c) [Weber et al. 2012], (d) [Lipman 2012] strict minimizer, (e) ASAP strict minimizer. The color maps and histograms show the quasiconformal distortion; note that ASAP does not optimize this measure directly, yet it optimizes it well.*

which require only two variables at for the vertex $v$. The resulting system has exactly the same structure as considered in Section 5. As a possible extension, one can generalize the relaxation-based strict minimization algorithm to handle free translations $t_{ij}$ (and thus determine their values), by introducing these as additional variables with global influence and adding iterative minimization of $L_\infty$ with respect to these variables to the process.

# B Proofs of Propositions

To prove Proposition 1, we prove an auxiliary proposition first.

Let $X$ be the set of solutions of the problem in (1), and $k$ the corresponding optimal value. Let the set of active constraints $A(\mathbf{h})$ for a point $\mathbf{h} \in X$ be the subset of indices $i \in I$, such that $D_i[\mathbf{h}] = k$. $X$ is a convex set, since the $D_i$ are convex functions. $X$ contains infinite solutions (we can interpolate between two points $\mathbf{f}, \mathbf{g} \in X$, treating the vertex coordinates as scalar functions).

**Proposition 5.** *For* $\mathbf{h} = (1-a)\mathbf{f} + a\mathbf{g}, 0 < a < 1$, $\mathbf{f}, \mathbf{g} \in X$, $A(\mathbf{h}) \subseteq A(\mathbf{f}) \cap A(\mathbf{g})$.

*Proof.* Consider a segment connecting two points $\mathbf{f}, \mathbf{g} \in X$, $\mathbf{h} = a\mathbf{f} + (1-a)\mathbf{g}$, for $0 < a < 1$. If some element $i \in A(\mathbf{h})$ is not in both $A(\mathbf{f})$ and $A(\mathbf{g})$, then either $D_i[\mathbf{f}] < k$ or $D_i[\mathbf{g}] < k$ (or both). Due to convexity, this means that $D_i[\mathbf{h}] \leq aD_i[\mathbf{f}] + (1-a)D_i[\mathbf{g}] < k$, which is a contradiction of $i \in A(\mathbf{h})$. □

**Proof of Proposition 1.**

*Proof.* It is sufficient to prove that the intersection of all active constraints sets of the (infinite) set of solutions in $X$ is not empty. If it is empty, there is also a finite subset of solutions of $X$ for which the intersection of sets of active constraints is empty: Because the set of constraints is finite, we can pick a single solution $\mathbf{x}$ for each constraint $i$, such that $A(\mathbf{x})$ does not contain $i$.

Suppose we have $m$ points $\mathbf{f}_i$ in $X$, such that the intersection of all $A(\mathbf{f}_i)$ is empty. Construct the midpoint $\mathbf{h}_1$ of the segment $(\mathbf{f}_1, \mathbf{f}_2)$. It has, by Proposition 5, an active constraint set contained in $A(\mathbf{f}_1) \cap A(\mathbf{f}_2)$. Repeat the process with the segment ($\mathbf{h}_1$,$\mathbf{f}_3$) and so on. Eventually, we construct a point $\mathbf{h}_{m-1}$ in $X$ with an active set $A(\mathbf{h}_{m-1})$ contained in the intersection of all $A(\mathbf{f}_i)$ which, by the supposition, is empty. But this contradicts the definition of $X$ (each solution must have an active constraint, i.e., an element with distortion touching the bound). We conclude that there must be a common active constraint for all of $X$. □

**Proof of Proposition 2.**

*Proof.* We prove by induction that any strict minimizer should have distortion value $z_j$ on the set $Ess(F_j)$ computed by the algorithm.

Suppose for any strict minimizer $\mathbf{f}^*$ and any $i \in F_{j-1}$, $D_i[\mathbf{f}^*] = k_i$, where $k_i = z_{j-1}$ is the distortion value associated with element $T_i$ by Algorithm 1, and $\mathbf{f}^* \in X_{j-1}$. We show that the same has to hold for $F_j$.

The statement above holds trivially for $j = 1$, as $F_0$ is empty, and we can define the set $X_0$ to be the set of all functions. Suppose $\mathbf{f}^*$ is not in $X_j$, i.e., not a solution of problem (1). As $D[\mathbf{f}^*]$ satisfies the bounds $k_i$ on all of $F_{j-1}$, $\mathbf{f}^*$ satisfies the constraints of (1). If $\mathbf{f}^*$ is not a minimizer of (1) in the $j$th iteration (the distortion on the unconstrained elements is greater than $z_j$), then for some element $i$ in $Ess(F_j)$, $D_i[\mathbf{f}^*] > z_j$. As a consequence, for any $\mathbf{g} \in X_j$, $\mathbf{g} \prec \mathbf{f}^*$, because $z_j^f < z_j^g$ and for all $\ell < j$, $z_\ell^f = z_\ell^g$ and $|I_{f,\ell}| = |I_{g,\ell}|$ . This contradicts the definition of a strict minimizer. As an element of $X_j$, $\mathbf{f}^*$ has to have value $z_j$ on $Ess(F_j)$. By induction, we conclude that for any strict minimizer, the distortion on each triangle is uniquely determined by Algorithm 1. □

We note that although the area factors in the definition of the sets $H_{f,z}$ are natural and allow the use of a uniform definition for continuous and discrete cases, they play no role in the algorithm, and hence, somewhat surprisingly, strict minimizers are not different from solutions of lexicographic minimax problems in the discrete case.

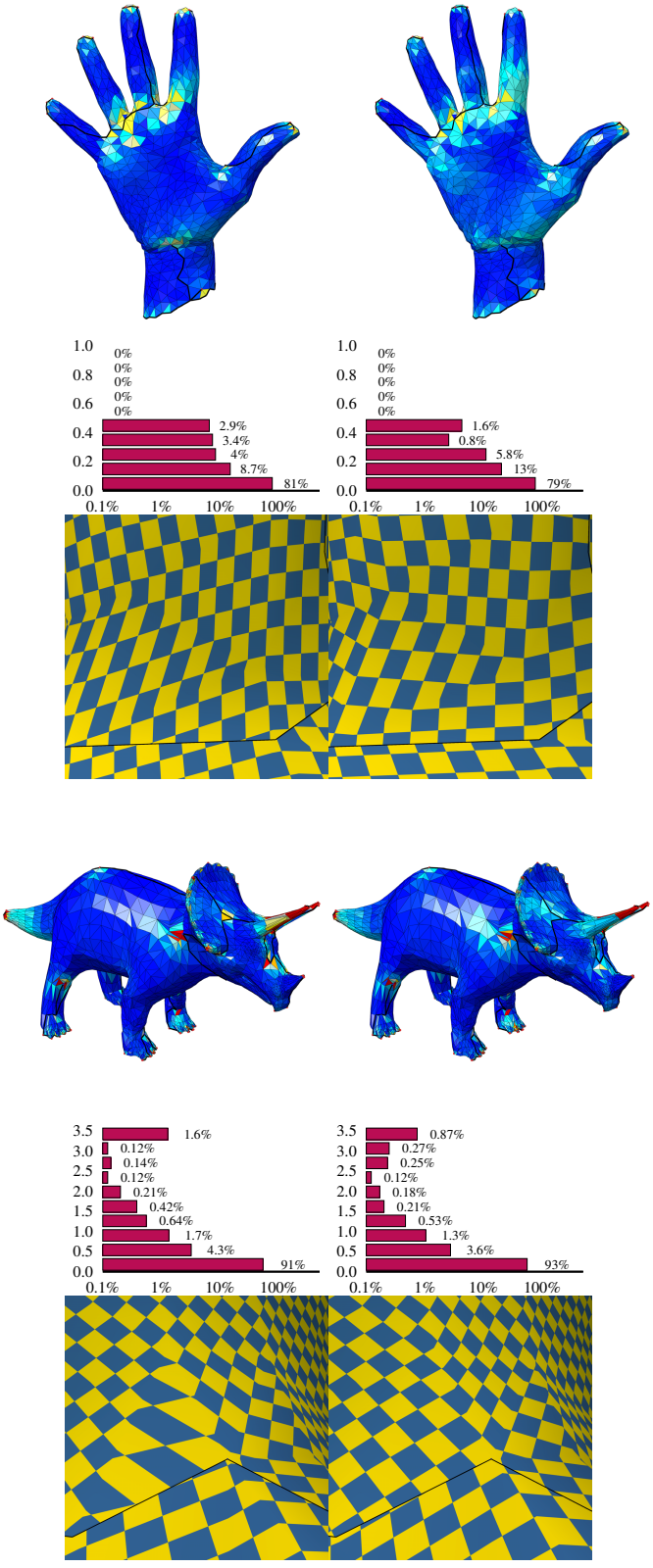



**Figure 16:** *[Aigerman and Lipman 2013] and a strict minimizer based on its algorithm and distortion measure.*

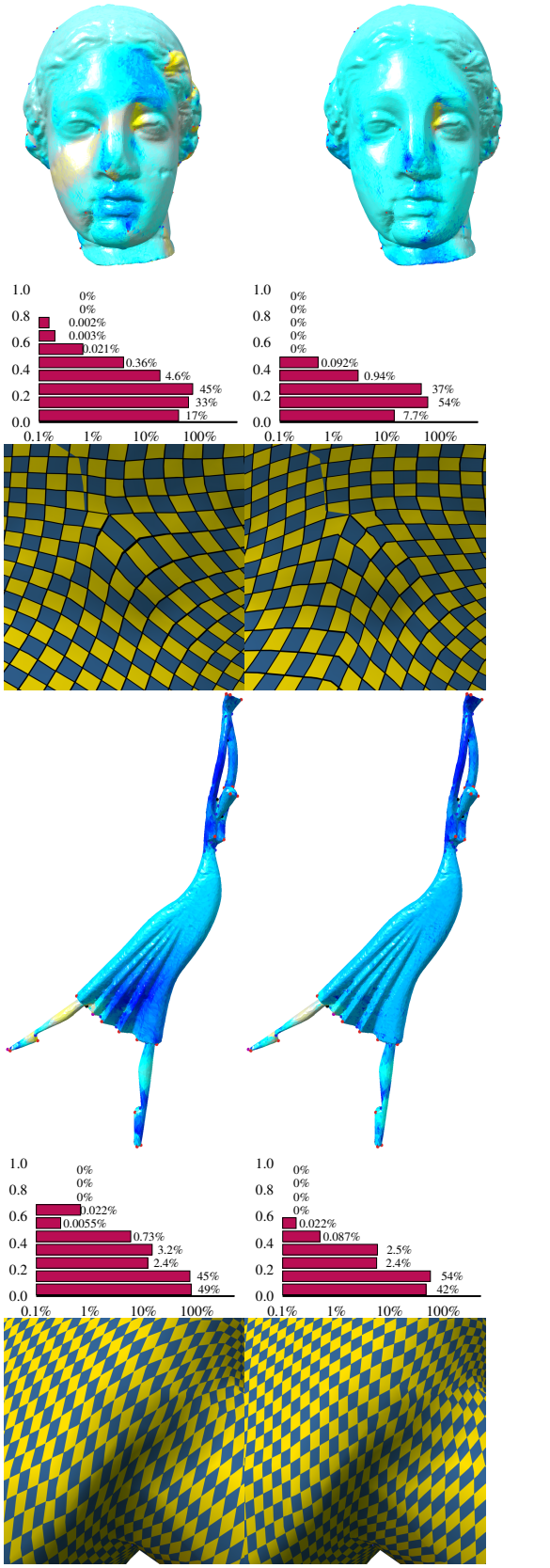



**Figure 17:** *field-deviation strict minimizer (left) compared to ARAP strict minimizer (right).*