Injective and Bounded Distortion Mappings in 3D

Noam Aigerman Yaron Lipman
Weizmann Institute of Science

Abstract

We introduce an efficient algorithm for producing provably injective mappings of tetrahedral meshes with strict bounds on their tetrahedra aspect-ratio distortion.

The algorithm takes as input a simplicial map (e.g., produced by some common deformation or volumetric parameterization technique) and *projects* it on the space of injective and bounded-distortion simplicial maps. Namely, finds a similar map that is both bijective and bounded-distortion. As far as we are aware, this is the first algorithm to produce injective or bounded-distortion simplicial maps of tetrahedral meshes. The construction of the algorithm was made possible due to a novel closed-form solution to the problem of finding the closest orientation-preserving bounded-distortion matrix to an arbitrary matrix in three (and higher) dimensions.

The algorithm is shown to have quadratic convergence, usually not requiring more than a handful of iterations to converge. Furthermore, it is readily generalized to simplicial maps of any dimension, including mixed dimensions. Finally, it can deal with different distortion spaces, such as bounded isometric distortion. During experiments we found the algorithm useful for producing bijective and bounded-distortion volume parameterizations and deformations of tetrahedral meshes, and improving tetrahedral meshes , increasing the tetrahedra quality produced by state-of-the-art techniques.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

Keywords: tetrahedral meshes, bounded distortion, conformal distortion, quasi-conformal, bijective mapping, simplicial maps

Links: DL PDF

1 Introduction

Mappings and deformations of tetrahedral meshes in three dimensional space ($\mathbb{R}^3$) are prevalent in computer graphics, geometric modeling and processing, medical imaging, physical simulations, and engineering. Nevertheless, the problem of producing injective and/or bounded distortion mappings of tetrahedral meshes remains mostly unsolved.

The goal of this paper is to introduce an algorithm that receives as an input a source simplicial mapping of a tetrahedral mesh (e.g., produced by existing deformation/mapping techniques) and approximates it with an injective bounded-distortion simplicial map.

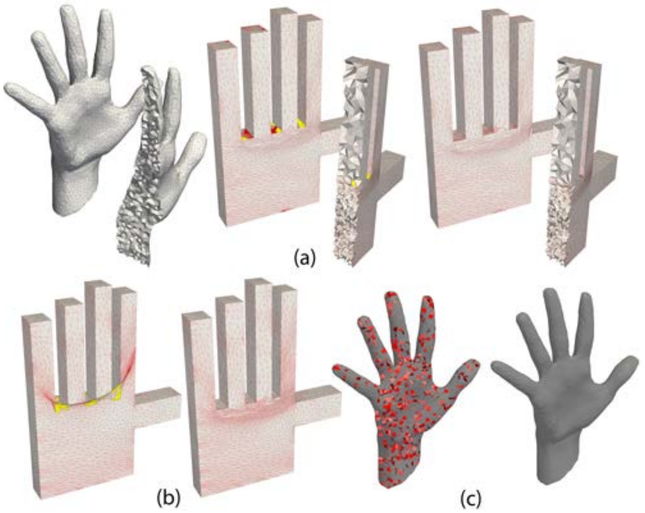


Figure 1: *The bounded-distortion projection algorithm presented in this paper can be used in several ways: (a),(b) approximating simplicial maps, in this case a discrete harmonic surface mapping ((b), left) and a discrete harmonic volumetric mapping ((a) middle), with a bijective bounded distortion simplicial map ((b), right, and (a) right). In (c) we use our algorithm to improve a tetrahedral mesh by eliminating tets with bad-aspect ratio.*

By bounded-distortion we mean that the aspect-ratio of the tetrahedra is not distorted too much. We call this procedure *projection* on the space of injective bounded-distortion simplicial maps. Figure 1(a) shows an example of a volumetric discrete harmonic map (middle, note the flipped tets in yellow and distorted ones in red) projected on the space of bounded-distortion bijective maps (right). Figure 2 shows another example of projecting a deformation of a bar.

The main challenge in producing injective and bounded-distortion mappings in three dimensions lies in the fact that the three-dimensional case is fundamentally different from its two-dimensional counterpart. In mappings of triangular meshes into the two dimensional plane, [Floater 2003] has shown that fixing a convex boundary and mapping each vertex to a convex combination of its neighbors leads to an injective mapping. However, as shown in [Floater and Pham-Trong 2006], these constructions fail to provide injective mappings in 3D. In [Lipman 2012] the space of injective and bounded-distortion mappings of triangular meshes into the plane is characterized, allowing to map triangular meshes into the plane injectively with bounded-distortion. Their technique depends heavily on the properties of complex numbers, and the fact that quadratic forms (i.e., quadratic homogeneous polynomials), like the determinant of a 2×2 matrix, can be easily brought into a canonical diagonal form. Unfortunately, since these properties are unique to two-dimensions, and no easy extension is known to three dimensions, the three dimensional case remains obscured.

In this paper we tackle the problem of constructing injective and bounded distortion maps in three dimensions. Our approach is based on several observations regarding the geometry of the collection of $d \times d$ bounded-distortion, orientation-preserving matrices.

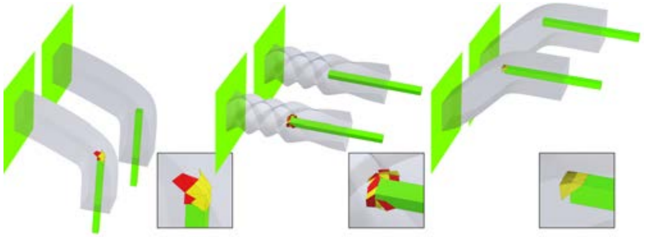


Figure 2: *As-rigid-as-possible deformations ([Sorkine and Alexa 2007], [Chao et al. 2010]) created by bending, twisting and translating a constrained rod inside a volumetric bar (bottom row) exhibit flipped and highly distorted tets, which are alleviated by our projection algorithm (top).*

The key tool we develop is a simple closed-form solution to the problem of finding the closest bounded-distortion and orientation-preserving matrix to a given arbitrary matrix.

Equipped with these observations we develop a rather simple algorithm that starts from a source map, and iteratively produces maps that are closer and closer to the space of bounded-distortion simplicial maps. Although we do not have a proof that the algorithm converges for all feasible instances, we do guarantee that upon convergence it outputs a locally injective bounded-distortion simplicial map. Furthermore, we prove that in case the boundary of the tetrahedral mesh is mapped bijectively, our algorithm produces a *globally* injective map. We believe this is the first algorithm that possesses a guarantee of this kind. Lastly, we demonstrate empirical numerical evidence, and provide a partial theoretical justification, showing that the convergence rate of this algorithm is quadratic.

The approach taken in this paper is general and allows several generalizations. First, it can readily be used for simplicial maps of any dimension, including the cases where the source dimension is not equal to the target dimension (e.g., Figure 1 (b)). Second, it is possible to replace aspect-ratio distortion with any type of distortion describable using linear inequalities of the singular values. For example, we demonstrate projection on bounded isometric distortion spaces, i.e., spaces with strict bounds on the singular values. Third, it can be used for tetrahedral mesh improvement, Figure 1 (c).

2 Previous work

3D deformations. There is a large volume of work dealing with deformations that aims at "shape preservation", that is loosely defined by controlling the distortion of the differentials. We provide only a partial list of relevant techniques here. A popular class of methods for 3D deformation is free-form deformation (FFD) which is based on linearly reproducing basis functions on regular grids [Sederberg and Parry 1986; Coquillart 1990]. Recently, FFD methods have been extended to general control polyhedra, and better shape preservation properties have been achieved [Floater et al. 2005; Ju et al. 2005; Joshi et al. 2007; Lipman et al. 2008]. Variational deformation methods [Ben-Chen et al. 2009] of tetrahedral mesh are not as popular as variational methods for surface deformation [Botsch and Sorkine 2008]. Nonetheless, most techniques can be adapted to the tetrahedral mesh setting. For elastic-type deformations, a popular energy is the as-rigid-as-possible energy [Sorkine and Alexa 2007]. In a related formulation, ARAP aims at minimizing the L^2 distance of the differential of the map to the rotation group [Chao et al. 2010]. Physically-based simulations (e.g., [Müller et al. 2002; Irving et al. 2004]) use integration of differential equations to guide the vertex displacements of tetrahedra. Weber et al., [2012] compute piecewise-linear approximations of extremal quasiconformal mappings. Johnen et al., [2012] show how to compute bounds on the jacobian of curvilinear finite element taking boundary straight-line elements to curved ones. In general, the above methods cannot guarantee injectivity and indeed often produce distorted/flipped tetrahedra. Our projection operator can produce good bounded-distortion approximations to the deformations produced by the methods mentioned above and others.

Tetrahedral mesh parameterizations and mapping is often done using Finite Elements (FEM) discretizations [Meyer et al. 2002] of harmonic mappings [Wang et al. 2004b; Wang et al. 2004a]. Li et al., [2007] approximate volumetric harmonic functions with the boundary-elements method. Xia et al., [2010] use the Green's function of the Laplacian to parameterize star-shaped volumes. Xu et al., [2012] approximate biharmonic volumetric functions with control over the derivatives across domains using the method of fundamental solutions. Gregson et al., [Gregson et al. 2011] map tetrahedral meshes into polycubes for building hex-meshes of volumes. [Cascón et al. 2009] Generate tet-meshes by mapping the boundary of a tetrahedral cube to a genus-zero surface. To the best of our knowledge, no existing method guarantees injective and/or bounded distortion mappings of tetrahedral meshes.

Tetrahedral mesh improvement. Tetrahedral mesh generation is a very active research domain [Owen 1998; Teng and Wong 2000; Eppstein 2001; Shewchuk 2012]. The focus in this paper is on the problem of tetrahedral mesh improvement, where given a tetrahedral mesh with fixed connectivity we ask how its vertices can be moved so that the *maximal* distortion of its elements can be lowered. Labelle and Shewchuk [2007] provide a tetrahedral mesh generation algorithm that is guaranteed to produce tets with angles bounded away from $0°, 180°$. Alliez et al., [2005] employ a variational approach that updates both vertex position and connectivity to reduce the Optimal Delaunay Triangulation (ODT) error functional, and the later work of Tournois et al., [2009] also adds sliver removal based on vertex perturbation. Klingner and Shewchuk [2007] introduce Stellar, a program that combines a collection of mesh improvement operations to build a practical algorithm for tetrahedral mesh improvement. Freitag and Knupp [2002] improve the condition number measured in Frobenius norm of the elements by a form of steepest-descent applied with heuristics to overcome the lack of smoothness of the objective function.

3 Preliminaries and problem statement

Mapping tetrahedral meshes is most commonly done by *simplicial maps*, that is maps that are affine over each tetrahedron (*tet*) and globally continuous across common faces. We study the subset of tetrahedral simplicial maps that possess desirable properties such as *injectivity* (local and global) and *bounded-distortion*. We start with setting notation and defining our goals.

A tetrahedral mesh $\mathbf{M} = \{\mathbf{T}, \mathbf{V}\}$ is a simplicial complex where $\mathbf{V} = \{\boldsymbol{v}_1 \dots \boldsymbol{v}_n\}$ is a set of vertices $\boldsymbol{v}_i \in \mathbb{R}^{3\times 1}$ (column vectors), and $\mathbf{T} = \{t_1 \dots t_m\}$ is a set of oriented tetrahedra. The most natural collection of maps that preserve the tetrahedral structure are simplicial maps, denoted here by $\mathcal{F} = \{\Phi\}$. A map $\Phi : \mathbb{R}^3 \to \mathbb{R}^3$ is said to be simplicial over $\mathbf{M}$ if each tetrahedron t_j is mapped via an affine transformation $\Phi|_{t_j}(\mathbf{x}) = A_j\mathbf{x} + \boldsymbol{\delta}_j$, where $A_j \in \mathbb{M} := \mathbb{R}^{3\times 3}$ is the (constant) *differential* of the transformation, $\boldsymbol{\delta}_j \in \mathbb{R}^{3\times 1}$ is the *translational* part, and $\mathbf{x} \in \mathbb{R}^{3\times 1}$ is a column vector representing a point in the tet t_j in the global coordinate system. Because it is piecewise-affine, Φ will be continuous if each affine transformation on tet t_j agrees with the affine maps of its neighbor tets over their common vertices. In fact, a (continuous) simplicial mapping $\Phi \in \mathcal{F}$ is uniquely determined by prescribing the images $\boldsymbol{w}_j = \Phi(\boldsymbol{v}_j)$ of each vertex, and extending linearly over each tet. The differential and translational parts of every affine map $\Phi|_{t_j}$ can be expressed as linear combinations of $\boldsymbol{w}_j$ by solving the following system of linear equations per tet once (e.g., at preprocess),

$$[A_j\ \boldsymbol{\delta}_j]\begin{pmatrix} \boldsymbol{v}_{j_1} & \boldsymbol{v}_{j_2} & \boldsymbol{v}_{j_3} & \boldsymbol{v}_{j_4} \\ 1 & 1 & 1 & 1 \end{pmatrix} = (\boldsymbol{w}_{j_1}\ \boldsymbol{w}_{j_2}\ \boldsymbol{w}_{j_3}\ \boldsymbol{w}_{j_4}) \tag{1}$$

where the tet $t_j = \langle \boldsymbol{v}_{j_1}, \boldsymbol{v}_{j_2}, \boldsymbol{v}_{j_3}, \boldsymbol{v}_{j_4} \rangle$. Throughout this paper $\boldsymbol{w} = \{\boldsymbol{w}_j\}$ is the only set of unknowns, and every time we write differential A_j or translation $\boldsymbol{\delta}_j$, they should be understood as constant linear combinations of $\boldsymbol{w}_j$, that is $A_j = A_j(\boldsymbol{w})$. This means, in particular, that the space of simplicial maps $\mathcal{F}$ of $\mathbf{M}$ can be identified with the linear space $\mathbb{R}^{3\times n}$, where each vector $\boldsymbol{w} \in \mathbb{R}^{3\times n}$ represents a unique simplicial map $\Phi = \Phi\boldsymbol{w}$.

In this paper we are interested in a subset of $\mathcal{F}$, namely simplicial maps that are locally injective and do not change too much the aspect-ratio of the tets. Both these properties are local and defined individually for every tet. Global injectivity is dealt with later on. Let us define these two properties in more detail.

Let $\Phi|_t(\mathbf{x}) = A\mathbf{x} + \boldsymbol{\delta}$ be an affine map of the tetrahedron t. Distortion of the affine map $\Phi|_t$ is defined in terms of the *signed singular values* of its matrix A, where signed singular values are a slight generalization of the classical singular values. The signed singular values of a matrix $A \in \mathbb{M}$ can be defined in terms of the *signed singular value decomposition* (SSVD) $A = U\Sigma V^T$, where $U, V \in \mathbb{M}$ are rotation matrices (in contrast to classical SVD where these are orthogonal matrices), and $\Sigma = \mathrm{diag}(\boldsymbol{\sigma}(A))$ is a diagonal matrix with the signed singular values on the diagonal $\boldsymbol{\sigma}(A) = (\sigma_1(A), \sigma_2(A), \sigma_3(A))$, $\sigma_1(A) \geq \sigma_2(A) \geq |\sigma_3(A)| \geq 0$. Note that $\sigma_3(A)$ can be negative, and in fact equals $\mathrm{sign}\,(\det(A))$ times the minimal singular value of A. $\sigma_1(A), \sigma_2(A)$ coincide with the top two singular values of A. The SSVD of a matrix A is computed by first computing the classical SVD of A and modifying the resulting U, Σ, V according to $\mathrm{sign}\,(\det(A))$.

The ratio $\frac{\sigma_1(A)}{|\sigma_3(A)|} \geq 1$ is the *aspect-ratio distortion* (also known in numerical analysis literature as the *condition number*) of the matrix A, and it expresses to what extent does A change the proportions of the tet it maps. We will want to keep this change bounded and require

$$\frac{\sigma_1(A)}{|\sigma_3(A)|} \leq K, \tag{2}$$

for some prescribed $K \geq 1$. Local injectivity is defined by requiring that the matrix A is *orientation-preserving*, i.e., $\det(A) > 0$, which is equivalent to

$$\sigma_3(A) > 0. \tag{3}$$

In case A is not orientation-preserving, it is orientation reversing, and one can imagine its effect on a tet as moving one vertex to the other side of the opposite face, like pulling a sock inside-out.

A main object in this paper will be the set $\mathbb{M}_K \subset \mathbb{M}$ of 3×3 matrices that satisfy both eqs. (2),(3), for a given K. We will also refer to this set as K-bounded-distortion matrices. A simplicial map $\Phi \in \mathcal{F}$ for which all its differentials are K-bounded-distortion, that is $A_j \in \mathbb{M}_K$, will be called a K-bounded-distortion simplicial map. We denote the space of K-bounded-distortion simplicial maps by $\mathcal{F}_K$.

We will measure distances between matrices A, B using the standard Frobenius norm $\|A - B\|_F$, where as usual $\|A\|_F^2 = \mathrm{tr}(A^T A)$. The distance between simplicial maps Φ, Ψ will be measured with

$$\mathrm{Dist}\,(\Phi, \Psi) = \left[\sum_{t_j \in \mathbf{T}} \|A_j - B_j\|_F^2 \, |t_i| \right]^{1/2}, \tag{4}$$

where, as before A_j is the differential matrix of the affine map $\Phi|_{t_j}$, and similarly we denote the affine map of Ψ restricted to tet t_j by $\Psi|_{t_j}(\mathbf{x}) = B_j\mathbf{x} + \boldsymbol{\gamma}_j$, and $|t_i|$ is the volume of tet t_j. This distance is the standard "rectangle" approximation of the smooth Sobolev seminorm $\int_{\mathbf{M}} \|D\Phi - D\Psi\|_F^2 \, d\mathrm{vol}$, where $D\Phi, D\Psi$ denote the differentials of (smooth) Φ, Ψ.

Algorithm 1: Projection on $\mathcal{F}_K$

```
Input:  Tetrahedral mesh $\mathbf{M} = (\mathbf{T}, \mathbf{V})$
        Source map $\Phi \in \mathcal{F}$ to project
        Distortion bound $K \geq 1$
Output: $K$-bounded-distortion map $\mathrm{Pr}\,(\Phi) \in \mathcal{F}_K$

$\Psi^0 = \Phi$;
while $\mathrm{Dist}\,(\Psi^{n+1}, \Psi^n) > \varepsilon$ do
    Compute SSVD $B_j^n = U_j \Sigma_j V_j^T$ for all $t_j \in \mathbf{T}$;
    Update:
        $\Psi^{n+1} = \mathrm{argmin}\,\mathrm{Dist}\,(\Psi\boldsymbol{w}, \Psi^n)$
            s.t.
            $\boldsymbol{w} \in \mathbb{R}^{3\times n}$
            and for every $t_j \in \mathbf{T}$,
            the diagonal $\mathbf{x} = (x_1, x_2, x_3)$ of $U_j^T B_j(\boldsymbol{w}) V_j$
            satisfies $x_k \leq K x_\ell,\ 1 \leq k, \ell \leq 3$
Return $\mathrm{Pr}\,(\Phi) = \Psi^{n+1}$;
```

Our goal in this paper is: given some map $\Phi \in \mathcal{F}$ find a close-by map $\Psi = \mathrm{Pr}\,(\Phi)$ in $\mathcal{F}_K$. We will call this procedure a *projection* (hence the notation $\mathrm{Pr}\,(\cdot)$) on the K-bounded-distortion space. Ideally, we would like to solve the following optimization problem for defining $\mathrm{Pr}\,(\Phi)$:

$$\mathrm{Pr}\,(\Phi) = \underset{\Psi \in \mathcal{F}_K}{\mathrm{argmin}} \ \ \mathrm{Dist}\,(\Phi, \Psi)\,. \tag{5}$$

Unfortunately, solving this problem globally is very challenging since the domain $\mathcal{F}_K$, to which Ψ is restricted, is composed of many copies of the space $\mathbb{M}_K$, which in itself is a rather convoluted non-convex set which poses a real challenge to work with: its definition involves singular values of a 3×3 matrix which are defined using the roots of a high-order polynomial in the entries of the matrix. The determinant related constraint (3) is a non-convex homogeneous cubic polynomial in the matrix entries which is also hard to control. Standard convexification of the space $\mathcal{F}_K$ (e.g., using the convex hull) would contain simplicial maps with arbitrarily-bad distortion, and hence a more elaborate convexification is required, as will be described next.

4 Algorithm

We tackle the challenge of optimizing eq. (5) by building a simple convex space of tetrahedral simplicial maps $\mathcal{F}_K^*(\Phi) \subset \mathcal{F}$ that provides an *approximation* to a part of $\mathcal{F}_K$ near Φ ($*$ will appear several times in this section, always to denote an approximation of a space). We use $\mathcal{F}_K^*(\Phi)$ to replace $\mathcal{F}_K$ in eq. (5). Since $\mathcal{F}_K^*(\Phi) \subset \mathcal{F}$ is not actually contained in $\mathcal{F}_K$ but only approximates it, our algorithm iterates until convergence:

$$\Psi^{n+1} = \underset{\Psi\boldsymbol{w} \,\in\, \mathcal{F}_K^*(\Psi^n)}{\mathrm{argmin}} \ \ \mathrm{Dist}\,(\Psi\boldsymbol{w}, \Psi^n)\,, \tag{6}$$

where we start with the source map $\Psi^0 = \Phi$. As will be shown, when the iterations converge, then $\mathrm{Dist}\left(\Psi^{n+1}, \Psi^{n}\right) \to 0$, and $\{\Psi_n\}$ converges to a K-bounded-distortion simplicial map with no flips, i.e. in $\mathcal{F}_K$. If the boundary of the domain is mapped bijectively, then the generated map is guaranteed to be a global bijection in 3D.

The functional $\mathrm{Dist}\left(\Psi \boldsymbol{w}, \Psi^{n}\right)$ as a function of $\boldsymbol{w}$, is convex and quadratic, and the space $\mathcal{F}_K^*(\Psi^n)$ is defined by several linear inequalities per tet. Hence, each iteration in eq. (6) requires solving a convex quadratic program with linear constraints. The algorithm has quadratic convergence properties; in practice it requires 4–10 iterations to converge. This excludes non-feasible cases (e.g., $\mathcal{F}_K$ is empty), in which case the algorithm does not converge. The algorithm is rather simple and can be implemented with any standard convex quadratic program optimizer. A full pseudo-code is presented in Algorithm 1, and no further information is needed in order to implement it. In order to understand the meaning and derivation of the particular linear constraints used in Algorithm 1 we need to better understand the geometry of the space of bounded-distortion matrices $\mathbb{M}_K$ and how the space $\mathcal{F}_K^*(\Phi)$ is constructed. This is done next.

4.1 Convex approximation spaces to $\mathcal{F}_K$

The key component in our algorithm is the space $\mathcal{F}_K^*(\Phi)$ that provides tractable approximations to $\mathcal{F}_K$. In this section we explain its construction and properties. Given a simplicial map Φ, the space $\mathcal{F}_K^*(\Phi)$ is defined with the following objectives in mind: 1) it should be a simple convex subspace of $\mathcal{F}$; 2) it should contain simplicial maps that are likely to be close to $\mathcal{F}_K$ and, as much as possible, to Φ. The construction of $\mathcal{F}_K^*$ was made possible due to novel observations about the space $\mathbb{M}_K$ which we detail next.

The main observation is that, surprisingly, although the set $\mathbb{M}_K$ is not convex, the problem of finding the closest matrix $\Pr(A) \in \mathbb{M}_K$ to an arbitrary matrix $A \in \mathbb{M}$, that is computing

$$\Pr(A) = \mathrm{argmin}_{B\, \in\, \mathbb{M}_K} \|A - B\|_F$$

turns out to have a closed-form solution. This is summarized in the following theorem proved in Appendix A,

Theorem 1. *Let $A \in \mathbb{M}$ be a non-zero arbitrary matrix. The closest K-bounded-distortion matrix $\Pr(A) \in \mathbb{M}_K$ is non-zero and has the form*

$$\Pr(A) = U X V^T,$$

where $X = \mathrm{diag}(\boldsymbol{x})$ is a diagonal matrix, and $A = U\Sigma V^T$ is the SSVD of A. Furthermore,

$$\boldsymbol{x} = \mathrm{argmin}_{\boldsymbol{y}\, \in\, \mathcal{T}_K} \|\boldsymbol{\sigma}(A) - \boldsymbol{y}\|_2\,, \qquad (7)$$

where $\mathcal{T}_K$ is the polyhedron

$$\mathcal{T}_K = \left\{ \boldsymbol{y} \in \mathbb{R}^3 \;\middle|\; y_1 \le K y_3\,,\ \ y_1 \ge y_2 \ge y_3 > 0 \right\}. \qquad (8)$$

Having this observation in mind we could have tried to define a map Ψ by setting its differentials to be $B_j = \Pr(A_j)$. Had such a simplicial map existed it would definitely be the global minimizer of eq. (5), and hence would be the desired $\Pr(\Phi)$. However, in general, the projected differentials $\Pr(A_j)$ are not compatible, that is, could not be used “as-is” to construct a continuous map, regardless of how one sets the translations $\boldsymbol{\gamma}_j$ of the affine maps $\Psi|_{t_j}$.

Therefore, we need to give some flexibility to the differentials B_j and search for compatible differentials in the *vicinity* of $\Pr(A_j)$. Let $A_j = U_j \Sigma_j V_j^T$ be the SSVD of the differential A_j. Then, we can try to fix the rotations U_j, V_j and look for a map Ψ with differentials of the form $U_j X_j^* V_j^T$, where X_j^* is “almost” diagonal, bounded-distortion matrix. We want to allow X_j^* to change from the optimal X_j defining $\Pr(A_j)$, but not stray too far from $\mathbb{M}_K$. As we will soon show, since the set of positive *diagonal* bounded-distortion matrices, which we denote as $\mathcal{D}_K$, is much simpler than the general $\mathbb{M}_K$, we are able to force X_j^* to stay close to the space of bounded-distortion matrices $\mathbb{M}_K$ by restricting it to belong to the subset $\mathcal{D}_K^*$ of diagonal K-bounded distortion matrices plus perturbations along the tangent space of $\mathbb{M}_K$.

These considerations lead to defining the convex sets $\mathbb{M}_K^*(A_j) \subset \mathbb{M}$, from which the differentials B_j of Ψ are sought. We define the main set of simplicial maps $\mathcal{F}_K^*(\Phi) = \{\Psi\}$ by requiring the differentials B_j of its members $\Psi \in \mathcal{F}_K^*(\Phi)$ to belong to $\mathbb{M}_K^*(A_j)$. We provide more details about the construction and properties of $\mathbb{M}_K^*(A_j)$ next. (for brevity we will drop the subscript j).

Construction of $\mathbb{M}_K^*(A)$. The sets $\mathbb{M}_K^*(A)$ will be built by taking the basic convex set $\mathcal{D}_K^*$ (to be defined shortly), and transforming it using pairs of rotation matrices U, V. That is, if the SSVD of A is $A = U\Sigma V^T$, then

$$\mathbb{M}_K^*(A) = U\mathcal{D}_K^* V^T = \left\{ UBV^T \middle| B \in \mathcal{D}_K^* \right\}.$$

The basic convex set of matrices $\mathcal{D}_K^*$ is defined to be

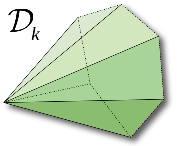
$$\mathcal{D}_K^* = \left\{ \begin{pmatrix} x_1 & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & x_2 & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & x_3 \end{pmatrix} \middle| \; \frac{x_i}{x_j} \le K\,, x_3 > 0 \right\}, \qquad (9)$$

where the off-diagonal entries $\varepsilon_{ij} \in \mathbb{R}$ are free. The main motivation behind its construction is that $\mathcal{D}_K^*$ can be written as a direct sum of two spaces $\mathcal{D}_K^* = \mathcal{D}_K \oplus \mathcal{E}$, where

$$\mathcal{D}_K = \left\{ \begin{pmatrix} x_1 & 0 & 0 \\ 0 & x_2 & 0 \\ 0 & 0 & x_3 \end{pmatrix} \;\middle|\; \frac{x_i}{x_j} \le K\;,\; x_j > 0 \right\}, \qquad (10)$$

and

$$\mathcal{E} = \left\{ \begin{pmatrix} 0 & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & 0 & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & 0 \end{pmatrix} \right\}.$$



Note that $\mathcal{D}_K \subset \mathbb{M}_K$ and consists of all positive diagonal bounded-distortion matrices. Geometrically, the diagonals of the matrices in $\mathcal{D}_K$ form a polyhedral cone in $\mathbb{R}^3$ with six infinite triangular faces, as shown in the inset. Furthermore, since $\mathbb{M}_K^*(A) = U\mathcal{D}_K^* V^T$ any $B \in \mathbb{M}_K^*(A)$ can be represented as:

$$U^T B V = D + E, \quad D \in \mathcal{D}_K, E \in \mathcal{E}. \qquad (11)$$

We will show the following properties of the sets $\mathbb{M}_K^*(A)$:

1. **Full dimension.** $\mathbb{M}_K^*(A)$ has the same dimension as $\mathbb{M}_K$.
2. **Closest bounded-distortion matrix.** The closest matrix in $\mathbb{M}_K$ to A, that is, $\Pr(A)$, is in $\mathbb{M}_K^*(A)$.
3. **Distance bound.** For all $B \in \mathbb{M}_K^*(A)$, $\mathrm{dist}(B, \mathbb{M}_K) \le \|A - B\|_F$, where $\mathrm{dist}(B, \mathbb{M}_K) = \min_{X \in \mathbb{M}_K} \|B - X\|_F$.
4. **Approximation of $\mathbb{M}_K$.** Every matrix $B \in \mathbb{M}_K^*(A)$ is on the tangent space of $\mathbb{M}_K$ calculated at $UDV^T \in \mathbb{M}_K$, where D is defined in eq. (11). That is, $B \in T_{UDV^T}\mathbb{M}_K$.

We show these properties next.

Property 1: Dimension of $\mathbb{M}^*_K(A)$. The set $\mathbb{M}^*_K(A) = U\mathcal{D}^*_K V^T$ is a subset of the nine-dimensional space $\mathbb{M} = \mathbb{R}^{3\times 3}$. Any matrix $D \in \mathcal{D}^*_K$ such that $D_{ii} < KD_{jj}$, for all i, j, can be perturbed (by adding small arbitrary values to all entries) and still not leave $\mathcal{D}^*_K$. This means that the interior of $\mathcal{D}^*_K$ (and therefore, by invariance to rotation, also that of $\mathbb{M}^*_K(A)$) is a (non-empty) open set in $\mathbb{M}$.

Property 2: Closest bounded-distortion matrix. This property is a consequence of Theorem 1. As noted, $\mathcal{D}_K$ is the set of all positive diagonal bounded-distortion matrices, and specifically contains all diagonal matrices whose diagonal is in $\mathcal{T}_K$. Theorem 1 now implies that $\mathrm{Pr}\,(A) \in U\mathcal{D}_K V^T$, and $U\mathcal{D}_K V^T \subset \mathbb{M}^*_K(A)$.

Property 3: Distance bound. Let $A = U\Sigma V^T \in \mathbb{M}$ be an arbitrary matrix. Property 3 asserts that the distance of any $B \in \mathbb{M}^*_K(A)$ to $\mathbb{M}_K$ is bounded from above by the distance of B and A. We will use the representation in eq. (11). Since $D \in \mathcal{D}_K \subset \mathbb{M}_K$, $UDV^T \in \mathbb{M}_K$, and we have that $\mathrm{dist}\,(B, \mathbb{M}_K) \leq \left\|B - UDV^T\right\|_F = \|E\|_F \leq (\|\Sigma - D\|_F^2 + \|E\|_F^2)^{1/2} = \|A - B\|_F$, where we used invariance to rotation of the Frobenius norm in the two equalities.

Property 4: Approximation of $\mathbb{M}_K$. This property ensures that a matrix $B \in \mathbb{M}^*_K(A)$ would be very (quadratically) close to $\mathbb{M}_K$ if $U^T BV$ is close to diagonal. This is the key to the algorithm's quadratic convergence.

To show this property we start with a matrix $B \in \mathbb{M}^*_K$, and decompose it again via eq. (11). We need to show that for $\mathrm{dist}\,(B, \mathbb{M}_K) = \mathcal{O}(\left\|B - UDV^T\right\|_F^2) = \mathcal{O}(\|E\|_F^2)$. For D in the interior of $\mathbb{M}_K$ this claim is clear as $\mathbb{M}$ is the tangent space for interior points. Hence, the main power of this analysis is in understanding this property for the boundary of $\mathbb{M}_K$, denoted by $\partial\mathbb{M}_K$. Since $\partial\mathcal{D}_K$ is piecewise smooth (it is a polyhedral surface and does not have tangent planes at the edges, see inset above), we are able to prove this property for "almost" all matrices $D \in \mathcal{D}_K$, namely those away from the edges of $\mathcal{D}_K$:

Theorem 2. *$\mathbb{M}^*_K(A)$ is a good approximation to $\mathbb{M}_K$ in the sense that for (almost) all $B \in \mathbb{M}^*_K(A)$, $\mathrm{dist}\,(B, \mathbb{M}_K) = \mathcal{O}(\|E\|_F^2)$, where E is defined in eq. (11). That is, the distance from B to $\mathbb{M}_K$ is of order of the* squared *norm of E.*

4.2 Properties of the projection algorithm

In each iteration, the projection algorithm in eq. (6) computes Ψ^{n+1} as the closest simplicial map to Ψ^n from the set $\mathcal{F}^*_K(\Psi^n) \subset \mathcal{F}$. The set of simplicial maps $\mathcal{F}^*_K(\Psi^n)$ by definition restricts the differentials B_j^{n+1} of candidates $\Psi^{n+1} \in \mathcal{F}^*_K(\Psi^n)$ to the convex spaces $\mathbb{M}^*_K(B_j^n)$, where B_j^n are the differential of Ψ^n. The four properties of the sets $\mathbb{M}^*_K(A)$ shown above can be used to justify the use of $\mathcal{F}^*_K(\Psi^n)$ in the projection algorithm.

Property 2 implies that the differentials B_j of simplicial maps in $\mathcal{F}^*_K(\Psi^n)$ are restricted to the set of matrices $\mathbb{M}^*_K(B_j^n)$ that contains the closest K-bounded-distortion matrix $\mathrm{Pr}\left(B_j^n\right)$ to the differential B_j^n of Ψ^n. Taking $B_j^{n+1} = \mathrm{Pr}\left(B_j^n\right)$ as the next iteration's Ψ^{n+1} differentials is the best we can hope for in terms of minimizing the distance $\mathrm{Dist}\left(\Psi^{n+1}, \Psi^n\right)$, but, as explained above, these differentials are generally not compatible (cannot be used without being altered for defining a continuous simplicial map). Property 1 shows that $\mathbb{M}^*_K$ has intrinsically the correct number of degrees of freedom (as $\mathbb{M}_K$).

Property 3 asserts that $\mathrm{dist}\left(B_j^{n+1}, \mathbb{M}_K\right) \leq \left\|B_j^{n+1} - B_j^n\right\|_F$, and summing this inequality (squared) over all tets results in

$$\left[\sum_{t_j \in \mathbf{T}} \mathrm{dist}\left(B_j^{n+1}, \mathbb{M}_K\right)^2 |t_j|\right]^{\frac{1}{2}} \leq \mathrm{Dist}\left(\Psi^{n+1}, \Psi^n\right).$$

The last inequality implies (somewhat surprisingly) that minimizing the distance to the previous map Ψ^n pushes the differentials B_j^{n+1} to the bounded-distortion space $\mathbb{M}_K$. In particular, convergence of the algorithm (i.e., $\Psi^n \to \Psi^\infty$) implies the limit map is bounded-distortion. Of-course, at the same time $\mathrm{Dist}\left(\Psi^{n+1}, \Psi^n\right)$ also tries to keep Ψ^{n+1} as close as possible to the previous Ψ^n, which serves as the basic motivation in the projection algorithm, namely of being similar to the source map.

Property 4 allows to improve the above inequality for small distances $\mathrm{Dist}\left(\Psi^{n+1}, \Psi^n\right)$. Theorem 2 implies[1] that the distance of B_j^{n+1} to $\mathbb{M}_K$ is asymptotically *quadratic* in the size of E_j, $\mathrm{dist}\left(B_j^{n+1}, \mathbb{M}_K\right) = \mathcal{O}(\|E_j\|_F^2)$. Similarly to the proof of Property 3 $\|E\|_F \leq \left\|B_j^{n+1} - B_j^n\right\|_F$, and hence

$$\mathrm{dist}\left(B_j^{n+1}, \mathbb{M}_K\right) = \mathcal{O}\left(\left\|B_j^{n+1} - B_j^n\right\|_F^2\right).$$

Therefore, it can now be shown that

$$\left[\sum_{t_j \in \mathbf{T}} \mathrm{dist}\left(B_j^{n+1}, \mathbb{M}_K\right)_F^2 |t_j|\right]^{\frac{1}{2}} = \mathcal{O}\left(\mathrm{Dist}\left(\Psi^{n+1}, \Psi^n\right)^{\mathbf{2}}\right).$$

This is a strong indication of quadratic convergence, which is experimentally verified in Section 5.

4.3 Bijective 3D simplicial maps

Constructing bijective mappings of tetrahedral meshes is mostly an open problem. In this section we show that mappings Ψ produced by Algorithm 1 are bijections in case the boundary of the tetrahedral mesh $\partial\mathbf{M}$ is mapped bijectively by Ψ. We will show that by proving the following theorem:

Theorem 3. *A simplicial map $\Psi \in \mathcal{F}$ that maps the boundary of **M** bijectively onto the boundary of a domain Ω, and that the matrices B_j of its affine maps $\Psi|_{t_j}$ satisfy eq. (3), that is $\sigma_3(B_j) > 0$ (or equivalently* $\det B_j > 0$*) is a global bijection between **M** and Ω.*

We give a proof in Appendix B. The proof uses degree theory to count the number of pre-image points to any point in the target space and show that points in the interior of Ω have exactly one pre-image, while points outside Ω have zero. Note that a proof for the planar case was given in [Lipman 2012], however, the method we use here is a generalization of that proof and can be used to prove injectivity of simplicial maps in higher dimensions.

4.4 Generalizations

Higher/lower dimensions. All the constructions in this paper can be readily applied to produce projections of simplicial maps in higher and lower dimensions, as-well as simplicial maps with mixed dimension such as mappings of surface meshes in 3D to the plane (parameterizations). In these cases, $\boldsymbol{w} \in \mathbb{R}^{d_t \times n}$, where d_t is the target space dimension, and the differentials $A_j, B_j \in \mathbb{R}^{d_t \times d_s}$, where d_s is the source space dimension. The SSVD $B_j^n = U_j \Sigma_j V_j^T$ has rotations $U_j \in \mathbb{R}^{d_t \times d_t}$, and $V_j \in \mathbb{R}^{d_s \times d_s}$. Noting these differences, Algorithm 1 can be applied.

[1] We assume the diagonals of the transformed differentials $U^T B_j^{n+1} V$ are away from the edges of the boundary cone $\partial\mathcal{D}_K$ (see Theorem 2 in Appendix A for details).

Projection on other spaces. Another useful generalization of our framework is to build projection operators on different spaces other than the K-bounded distortion simplicial maps $\mathcal{F}_K$. In particular, when examining the construction in the beginning of this section it can be seen that one can use any space that is describable with linear inequalities of singular values. This can be done by replacing $\mathcal{D}_K$ with a different polyhedron. One example could be the space of simplicial maps with bounded isometric distortion, where we restrict $\max \boldsymbol{\sigma} \leq S$, and $\min \boldsymbol{\sigma} \geq S^{-1}$. Note that in this case the cone $\mathcal{D}_K$ is replaced with the cube: $\left\{ \mathrm{diag}(\mathbf{x}) \;\middle|\; S \geq x_i \geq S^{-1} \right\}$.

4.5 Further implementation details

We represent a simplicial map $\Psi = \Psi_{\boldsymbol{w}}$ by a vector $\boldsymbol{w} \in \mathbb{R}^{3\times n}$ which encodes the target positions of the vertices of the mesh $\mathbf{M}$. The differentials $B_j = B_j(\boldsymbol{w})$ are represented as linear combinations of entries of $\boldsymbol{w}$ as can be computed from eq. (1). The functional $\mathrm{Dist}\left(\Psi_{\boldsymbol{w}}, \Psi^n\right)$ in Algorithm 1 is quadratic in $\boldsymbol{w} \in \mathbb{R}^{3\times n}$ and can be explicitly written by plugging in the linear expressions for $B_j(\boldsymbol{w})$ in eq. (4), considering the matrices B_j^n as constant. Each iteration of the algorithm amounts to solving one convex quadratic program with linear inequalities. We have used the sparse interior-point quadratic program solver supplied with Matlab2012a, with a termination tolerance and constraint-violation tolerance of 10^{-14}. All other parameters were set to default.

Inequality constraints. Let us show that the constraints $\Psi_{\boldsymbol{w}} \in \mathbb{M}_K^*(\Psi^n)$ in eq. (6) are equivalent to the inequalities in Algorithm 1. By definition of $\mathbb{M}_K^*(\Psi^n)$ we have by (11) that $U_j^T B_j(\boldsymbol{w}) V_j = D_j + E_j$, where $D_j \in \mathcal{D}_K$ and $E_j \in \mathcal{E}$. This is equivalent to that the diagonal of $U_j^T B_j(\boldsymbol{w}) V_j$ is in $\mathcal{D}_K$ which are exactly the constraints in Algorithm 1, with one exception, in Algorithm 1 we do not have an inequality for excluding the case $\mathbf{x} = (0, 0, 0)$. Although it can be added (e.g. $x_3 \geq \varepsilon$, with $\varepsilon > 0$ a small constant), the time complexity of the algorithm is related directly to the number of constraints and we wish to avoid adding unnecessary constraints. For any non-zero matrix A, $\mathrm{Pr}\,(A)$ is always non-zero (see Theorem 1), therefore, in practice, zero projection never happens, and neither did we encounter zero differentials. In case they do appear, it is possible to rerun that iteration after adding constraints of the type mentioned above to avoid zero diagonal.
Another modification to Algorithm 1 that improves the time complexity of the algorithm is to keep only the constraint $x_1 \leq K x_3$ among the inequalities. This reduces the number of inequalities by a factor of six and provides a significant speed up. The justification here is that the distance of $\boldsymbol{\sigma}(A)$ to the half-space $y_1 \leq K y_3$ is always *maximal* among all the half-spaces defining the cone $\mathcal{D}_K$. (This stems from the fact that the vector $\boldsymbol{\sigma}(A)$ is already ordered.) This means that in each iteration, and for each tet, we always make sure to be inside the farthest half-space (defining $\mathcal{D}_K$) from $\boldsymbol{\sigma}(B_j^n)$. Upon convergence this means that we end up in $\mathcal{D}_K$, and the guarantees of the algorithm still hold with this modification. In practice we have not noticed degradation in the algorithm's performance due to this modification, on the contrary, in some cases it improved convergence mainly due to the fact that it enlarges the feasible domain. All our experiments are done with this modification incorporated.

Positional constraints can be easily incorporated in Algorithm 1 by forcing some of the unknowns in $\boldsymbol{w}$ to user prescribed positions, that is $\boldsymbol{w}_i = \boldsymbol{w}'_i$ for some subset of vertices $\boldsymbol{v}_i \in \mathbf{V}' \subset \mathbf{V}$.

5 Experiments

Algorithm 1 can be applied and used in many scenarios in geometry processing and modeling, due to its generic ability to approximate a given (source) simplicial map Φ with an injective K-bounded-distortion map. The only parameter of this algorithm is K, and it provides a clear trade-off between approximation power and distortion: if K is small the target space $\mathcal{F}_K$ is smaller and the approximated projection $\mathrm{Pr}\,(\Phi)$ is farther from the source map Φ. If K is high, $\mathcal{F}_K$ is more flexible and yields a better approximation to Φ.

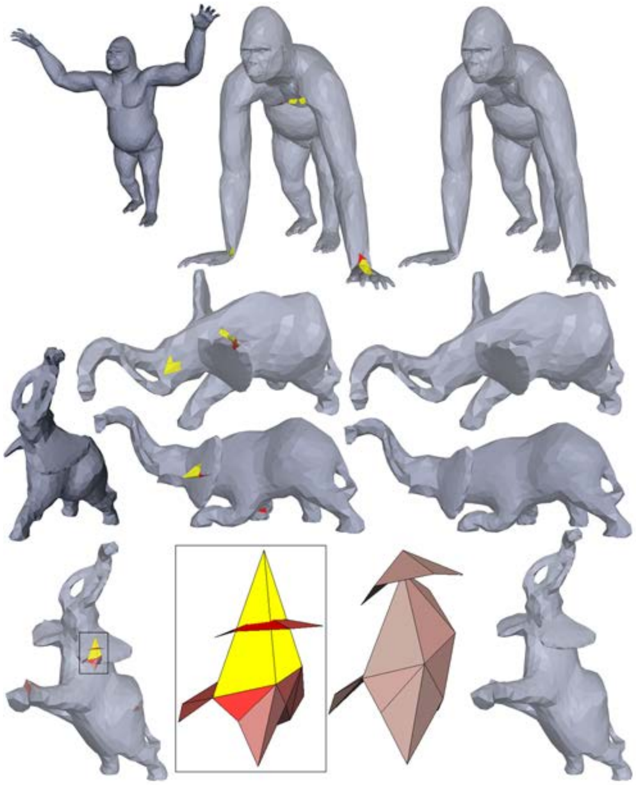


Figure 3: *Approximating ARAP deformations with bounded-distortion maps. Note the yellow and red tets showing flipped and highly distorted tets (we render ones with aspect-ratio distortion > 25) in ARAP, and our bounded-distortion approximation. In the bottom row, a blow-up of a cluster of flipped and highly distorted tets before and after the projection is shown. Rest-pose models are shown in dark-grey.*

In what follows we overview a series of applications of the projection algorithm, and discuss some numerical experiments. Throughout this section aspect-ratio distortion is colored in red (low distortion in gray, high distortion in red), and flipped elements (tets, and later, triangles) in yellow.

5.1 Deformation

The simplest and most immediate application of our algorithm is projection of tetrahedral mesh deformations on the bounded-distortion space. We examined projections of deformations created by three different methods that span fairly well the range of existing volume deformation techniques. In all the following deformations we have used a bound of $K = 5$, except for the biharmonic example where we use $K = 7$. In the examples in which positional constraints were used to generate the deformation (i.e. the ARAP and Biharmonic examples) our projection is performed with the same positional constraints.

As-Rigid-As-Possible (ARAP) deformation is a variational finite-elements-based technique that strives to keep the differentials of the map as close as possible to rotations [Chao et al. 2010]. As

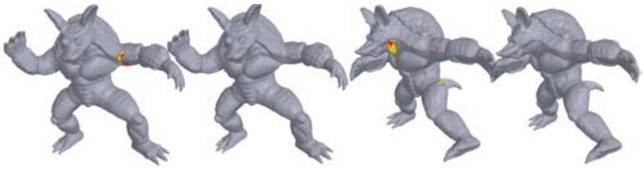
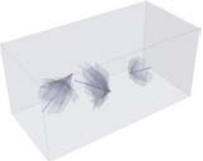


Figure 4: *Deformations of the Armadillo using Mean Value Coordinates, and their bounded-distortion approximation. Flipped and highly-distorted (> 25) tets are highlighted.*

such, it generally resists flips and high aspect-ratio distortion, but cannot avoid it. In practice, tets present high distortion and/or flips during deformation, usually in areas close to the constrained vertices, as shown in Figure 2. In Figure 3, the gorilla's palms were constrained to touch the floor while its waist was held in place. As a result, ARAP created flips (see yellow tets) and high distortion (red tets) near the palms and in the torso. Our algorithm is able to approximate this deformation well while removing these artifacts (right). The distance from our projection to the source deformation was 0.03 in this case, where the mesh $\mathbf{M}$ has unit volume (this will be the convention with of all our models). The elephant model was deformed to three different positions. A blow-up on the bottom row shows a flipped tet intersecting a highly distorted tet in an ARAP deformation, and the same tets after the projection (right). Note how the intersection is resolved. The distances of the projected map to the source map for the elephant were 0.19, 0.12 and 0.2 respectively.

Mean Value Coordinates (MVC) is a free-form deformation technique that defines a smooth volumetric deformation based on the deformation of a surrounding control mesh [Ju et al. 2005; Floater et al. 2005]. Although MVC defines a smooth deformation it does not have control over distortion and flips. Figure 4 shows projection of two deformations of the Armadillo onto the bounded distortion space. Note again the visual approximation quality of the bounded distortion map; The projection distances are 0.04 and 0.06. The left Armadillo had initial maximal aspect-ratio distortion of 520 and 10 flipped tets. The right Armadillo had an initial distortion of 1000 and 111 tets were flipped.

Biharmonic volume deformation provides a smooth deformation [Jacobson et al. 2011], but similar to other techniques it cannot guarantee absence of flips or high distortion. In Figure 5 we fixed the boundary of a box and moved 3 vertices in the interior (see inset) to create a strong biharmonic deformation with a maximal distortion of 230 and 67 flipped tets. We projected the biharmonic map with $K = 7$. The result is shown in Figure 5. At each row we present 3 cross cuts into the mesh at different heights, presenting the source map on the top row and the projected map at the bottom. Flips and lack of bijectivity are recovered by a rather significant change to the map. In this case the distance of the projection to the source map was 0.46. Note that Theorem 3 assures a global bijective map.



5.2 Parameterization

The second application of our projection algorithm is bijective volume parameterizations of tetrahedral meshes to polycube-type domains. On each model we manually marked the polycube curve structure as shown by blue lines on the meshes in Figure 6. As a source mapping we built a smooth map and then projected it using our algorithm (results shown in Figure 6). The source mapping Φ is generated in a few stages all involving our algorithm. We first build a bijective mapping of the boundary surface mesh to the boundary of the polycube. This is done by mapping the boundary to the polycube faces using discrete harmonic mapping [Eck et al. 1995], and then projecting this two dimensional simplicial map on the bounded distortion space using our algorithm, as displayed in Figure 1(b). We provide more details on projections of simplicial maps of different dimensions (like this one) later in this section. Once we have the initial bijective, bounded-distortion map of the boundaries we map the interior with discrete volumetric harmonic mapping again to achieve our source map Φ. We found it useful to allow each boundary vertex of $\mathbf{M}$ that is mapped to faces and/or edges of the polycube to move on their faces/edges to allow lower distortion. This is achieved by substituting the strict positional constraint with the following *linear* constraints in Algorithm 1: each boundary vertex is confined to move only on the infinite plane which supports its target polycube face. Vertices on edges are confined to two planes simultaneously. Corner vertices are mapped to corners of the polycube. Figures 1,6 depict a collection of bijective tetrahedral mappings. In the inset we display the polycubes' isolines mapped to the original volume, before and after projection.

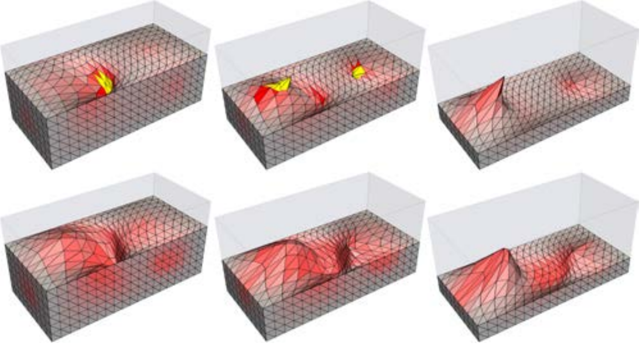
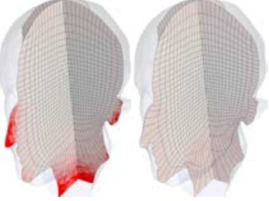


Figure 5: *Biharmonic deformation of the volume inside a box. The boundary is constrained to stay in place and 3 vertices are constrained to different positions in the interior. Top row shows the source biharmonic map, cut at three different heights. Bottom row shows the respective bijective bounded-distortion approximation.*



Theorem 3 guarantees that in case the boundary of $\mathbf{M}$ is mapped bijectively to the boundary of the polycube, then the projected map Ψ is a global bijective map. We note that if we allow the vertices to move on the supporting planes, Theorem 3 can still be used by noting the invariance of the degree to homotopy. The K used for the different models are: duck - 4; max planck - 6; bimba - 8; sphinx - 10; rocker - 20; hand - 20.

Mesh connectivity issues. It is important to note that it is impossible to construct a bijective mappings onto polycubes in case there exist "dividing" edges\faces, i.e., interior edges or faces whose vertices lie on the boundary. Therefore, for the application described above we subdivide as necessary to cancel any such tets.

5.3 Tetrahedral mesh improvement

Another application of our algorithm is tetrahedral mesh improvement, by moving the mesh's vertices so that all tets have bounded aspect-ratio. We emphasize that we do not have a convergence proof, but upon convergence, the algorithm is guaranteed to generate a tetrahedral mesh comprised of tets with bounded aspect-ratio.

We achieve this by casting the mesh improvement algorithm into

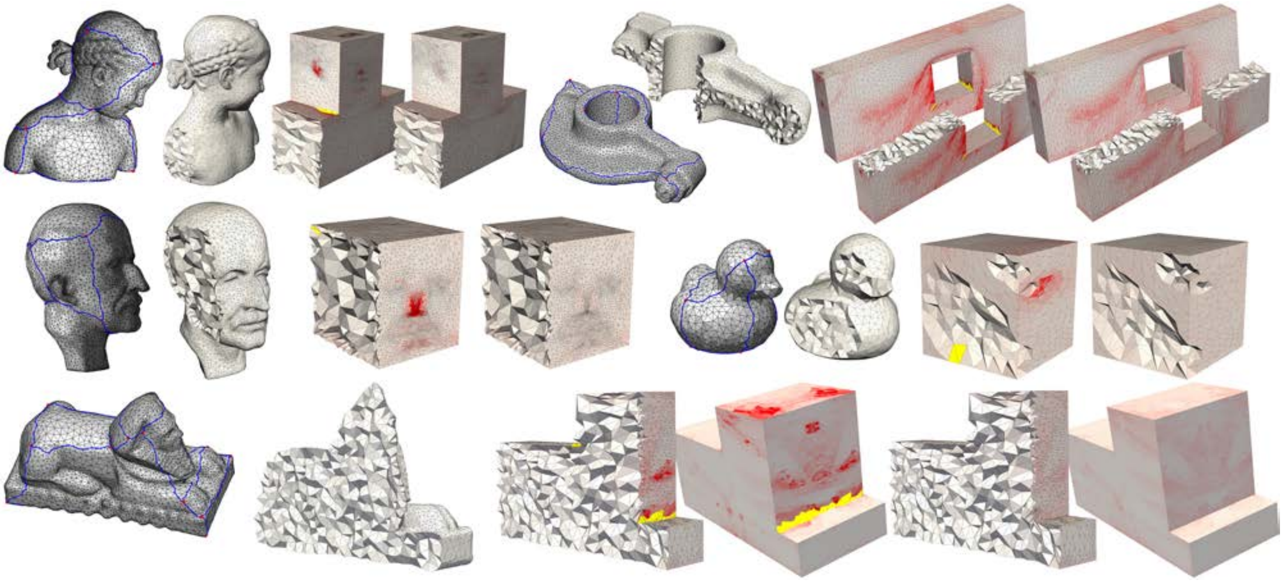
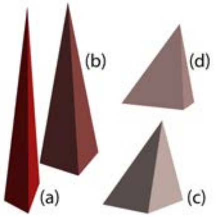


Figure 6: *Five volumetric meshes mapped bijectively with bounded distortion to polycubes. For each mapping we show: the polycube edges and vertices as they were marked on the original meshes (edges in blue, vertices in red), the source discrete harmonic mapping (note flipped tets in yellow and high distorted ones in red) and its bijective bounded-distortion approximation.*

the problem of finding a bounded-distortion mapping, similar in spirit to the planar mesh improvement in [Lipman 2012]. The idea is to start with a collection of m (remember that we have m tets in $\mathbf{M}$) positively-oriented perfectly-regular tets (defined by having all edges of equal length) and to look at the piecewise-affine map taking these perfect regular tets to the tets of the mesh $\mathbf{M}$. Then, we note that Algorithm 1 can still be used as-is to look for a vector $\boldsymbol{w} \in \mathbb{R}^{3\times n}$ such that the differentials $B_j(\boldsymbol{w})$ over each regular tet will have bounded distortion and no flips. This will guarantee a tetrahedral mesh with tets of bounded aspect-ratio.

We applied this procedure to a collection of 15 tetrahedral meshes produced by three state-of-the-art algorithm: Variational Tetrahedral Meshing (VTM) [Alliez et al. 2005], Optimal Delaunay Triangulation with sliver removal techniques using the CGAL software library [Cgal], and Aggressive Tetrahedral Mesh Improvement (STE) [Klingner and Shewchuk 2007]. The results are summarized in Table 1 (see also Figure 7 for visualization of 5 out of the 15 meshes produced): the table logs the initial (before) and final (after) maximal aspect-ratio of the elements (calculated w.r.t. to a perfectly-regular tet), and the minimal and maximal dihedral angles of the mesh, which are one of the most common criteria of mesh quality. The algorithm significantly improves the worst aspect-ratio of the tetrahedra in all meshes. Furthermore, the algorithm improves the maximal dihedral angles of all meshes, and the minimal dihedral angles of all meshes but one - the dragon. Indeed, the aspect-ratio and dihedral angles are not equivalent criteria. However, in some sense bounded aspect-ratio is stronger, as bounded aspect-ratio implies the dihedral angles are bounded away from $0^\circ, 180^\circ$, but not vice-versa - for example, consider a spire tetrahedron such as tet (a) in the inset. The inset shows a visual comparison of the tet with the worst aspect-ratio distortion in the dragon, shown in (a) and the same tet after projection in (b). Although (a) has a large aspect-ratio, it has a large minimal dihedral angle. (d) shows the tet that had the smallest minimal dihedral angle after the projection, and the same tet from the original dragon's mesh in (c). The tradeoff here seems clear: to improve their dihedral angles, tets can be stretched into spires, at the price of enlarging their aspect-ratio. Lastly, let us note that while we do not constrain the boundary at all, hard positional constraints on the boundary are optional and can be incorporated at the price of a larger K - in our experiments constraining the boundary usually raises the feasible K to 10.



mesh	K before	K after	dihed before	dihed after
bimba	44.6	**5**	(8,168)	(**16**,**148**)
elephant	19.3	**5**	(8,167)	(**16**,**148**)
horse	21	**5**	(10,164)	(**16**,**148**)
ramesses	23.7	**5**	(7,168)	(**16**,**148**)
rocker	13	**5**	(10,163)	(**16**,**148**)
sphinx	19.6	**5**	(8,167)	(**16**,**148**)
duck	12	**5**	(10,163)	(**16**,**148**)
gorilla	61	**5**	(2,173)	(**16**,**148**)
hand2	25	**5**	(8,166)	(**16**,**148**)
tooth	11.3	**5**	(11,164)	(**16**,**148**)
hand	10	**5**	(9,162)	(**16**,**148**)
skull	117	**6**	(0.8,178)	(**14**,**153**)
elephant	22.4	**5**	(15,157)	(**18**,**147**)
max	14.8	**5**	(21,151)	(**21**,**147**)
dragon	8	**5**	(**31**,140)	(28,**139**)

Table 1: *The results of our mesh improvement algorithm applied to meshes produced by CGAL [Cgal], VTM [Alliez et al. 2005], and STE [Klingner and Shewchuk 2007] (grouped in this order in the table). We display maximal aspect ratio (K) before and after improvement, as well as minimal and maximal dihedral angles. Note that aspect-ratio is improved for all cases, and dihedral angles are improved for all cases but one.*

5.4 Extensions and Generalizations

Simplicial maps in other dimensions. As mentioned in Section 4.4, our algorithm can be applied seamlessly to the problem of approximating simplicial maps in any dimension. An example application is surface parameterization, i.e. flattening a surface mesh into

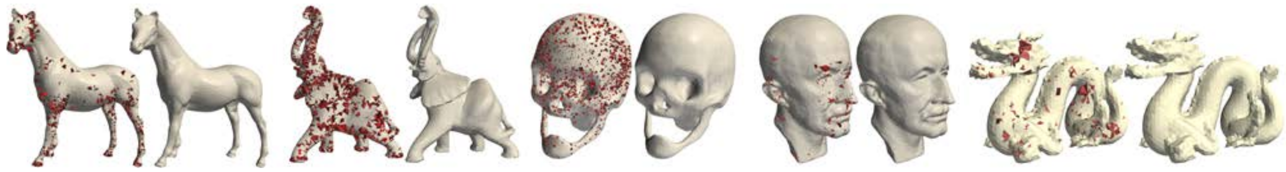


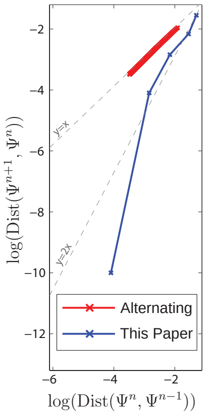
Figure 7: *Results of our volumetric mesh improvement. For each mesh we show the source mesh (produced by CGAL, VTM, and STE, see text for details) with all tetrahedra with aspect-ratio above the prescribed bound shown, and the results of our algorithm (in which all tets' aspect ratio is below the threshold and therefore no red tets are shown).*

the 2D plane, where the bijectivity and bounded-distortion guarantees remain valid. In Figure 8, we reproduced such a result from [Lipman 2012]. We took the same input (ARAP parameterization by [Liu et al. 2008]) and projected it using our algorithm and got a similar result as [Lipman 2012]. As mentioned in Subsection 5.2 (see e.g., Figure 1), we also solve a surface parameterization problem (mapping a patch of the boundary of a tet-mesh to a face of the polycube) for each polycube we produce.

Bounded isometric distortion. Our framework can be adapted to project maps on different bounded-distortion spaces (see Subsection 4.4). In the inset we display a projection of an ARAP deformation of a dinosaur model, this time on the bounded-isometric distortion space, bounding the singular values strictly between $\frac{1}{3}$ and 3. This space is a subset of the $K = 9$ bounded-aspect ratio space.

5.5 Numerical properties and alternatives

Alternative optimization techniques can be considered by using the core observation of Theorem 1. For example, the closed-form solution of a projection of a matrix on the bounded-distortion space can be used in an alternating projections (AP) algorithm for optimizing eq. (5) in the spirit of [Liu et al. 2008], i.e., first find (via Theorem 1) the closest bounded-distortion differentials, second solve a Poisson equation to find the closest differentials that constitute a continuous map, and repeat the two steps until convergence. We have implemented such an algorithm and tested it. Its main drawback is its bad convergence rate. Although in this alternative optimization method each iteration is much faster, as it requires solving a linear system with a constant matrix and not a quadratic program, the number of iterations required is huge. In the inset we show a log-log plot (in base 10) of Dist $\left(\Psi^{n+1}, \Psi^{n}\right)$ as a function of Dist $\left(\Psi^{n}, \Psi^{n-1}\right)$. As the dashed lines illustrate, AP's graph lies on the line $y = x$ while ours lies on the line $y = 2x$, verifying AP converges linearly while ours converges quadratically. Indeed, our algorithm penetrated the bounded-distortion space after 5 iterations, taking a total of 310 seconds. In contrast, we stopped the AP algorithm after 500 iterations which took 6000 seconds, as it was not close to convergence. In fact, when we stopped it, the AP algorithm had already strayed farther away from the original map than the distance of our algorithm's final projected map to the original map, hinting that our algorithm also provides better approximations of the source map.



#V	#T	#iter	total time	1 iter time	SVD time	quadprog solve time	setup time
125	384	6	1.8	0.2	0.05	0.06 (30%)	0.6
512	2K	6	10	1.2	0.25	0.6 (50%)	2.8
1K	4K	7	26	3	0.53	1.5 (50%)	5
2K	8K	6	48	7	1	3.8 (54%)	6
4K	20K	6	147	20	2.5	14 (70%)	27
8K	40K	6	325	45	5	33 (73%)	55
16K	83K	6	935	130	10	110 (84%)	130

Table 2: *Timings in seconds of our algorithm applied to the same map at different mesh resolutions.*

Timings. Following are typical timings of each application presented: projecting a deformation of a mesh with 20k tets, 5.2k vertices converged after 4 iterations for a total of 80 seconds. Mapping a mesh with 45k tets and 11k vertices to a polycube converged after 7 iterations, in 407 seconds. Improving a mesh with 156k tets and 37k vertices converged after 5 iterations, in half an hour. Table 2 details the timings of our algorithm applied to the map $f(x, y, z) = \left(x, y + x^2 \sin(5\pi x), z + x^2 \cos(5\pi x)\right)$, sampled at several mesh resolutions on the unit cube. As can be seen, asymptotically the running time is affected mainly by the time it takes to solve the quadratic problem and compute the SSVD of all differentials. All results were timed using a single-process on a 3.40GHz Intel-i7 CPU.

Comparison to [Lipman 2012] When the target space is 2D, our algorithm can be compared to [Lipman 2012]. While our optimization domain is an approximation of a part of $\mathcal{F}_K$, Lipman optimizes over a convex subset of $\mathcal{F}_K$. In Figure 9 we display the results of our algorithm on data sets from [2012]. In most cases the differences are minute, and are more noticeable in areas where the initial map is far from satisfying the bounded-distortion properties. The running times of both methods are comparable.

6 Concluding remarks

A simple but theoretically principled method for approximating a given simplicial map of a tetrahedral mesh with an injective bounded-distortion map is introduced. In case the boundary of a mesh is mapped bijectively then the presented algorithm outputs a global bijection. The algorithm can be used to produce bounded-distortion approximations of deformations created by popular deformation algorithms, to produce bijective 3D parameterizations, and to improve worst aspect-ratio of tetrahedral meshes. The algorithm can be extended to simplicial mappings in any dimension (e.g. surface parameterization), and to other types of bounded-distortion spaces, like bounded-isometric distortion.

The main drawbacks of our algorithm are two-fold: first, each iteration requires solving a quadratic program and although only a handful of such iterations are required, this limits the scalability of

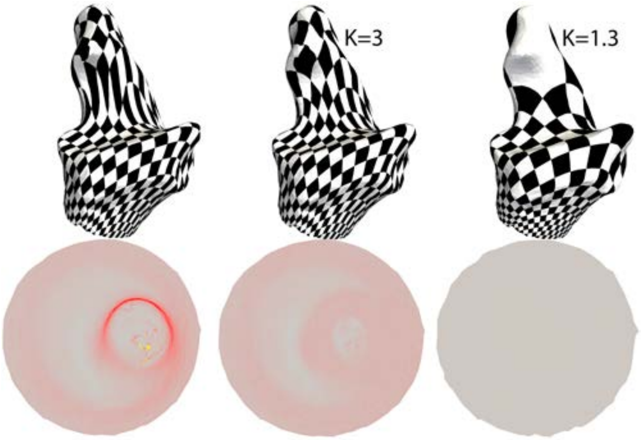


Figure 8: *A surface mesh of a bone model parameterized to the plane with an ARAP [Liu et al. 2008] initial guess (left) and projected on the bounded-distortion space with $K = 3$ (middle), and $K = 1.3$ (right). Note the alleviated distortion and flips (compare to Figure 6 in [Lipman 2012]). As expected, a higher K leads to better approximation, while pushing K as low as possible provides an approximation of a conformal map, farther away from the initial ARAP source map.*

the algorithm. One possible solution is to constrain only differentials that are not deep inside the bounded-distortion space; this will reduce the number of constrained differentials drastically. Another speedup can be achieved by exploring different solvers. Second, the question of convergence and feasibility is still open. One of our future goals is to show that in case the problem is feasible (for some given bound K, and positional constraints) the algorithm is guaranteed to converge, under some additional assumptions, e.g. that the initial map is "close enough" to being bounded-distortion. Lastly, it should be noted that in case the source map was derived using some physical equation or variational principle the projected map does not necessarily adhere to the same governing equation.

This paper is a first step in producing mappings and deformations in 3D with guarantees. There are many more algorithms and geometry-processing problems we wish to tackle with the new method presented here. First, it can be applied directly to existing algorithms like Gregson's et al., [2011] hex generation to achieve a guaranteed bound on the distortion of the hex elements. It can be applied to 3D surface mappings problems where the challenge is to restrict the target domain to be a manifold. We also believe it holds potential to 3D matching and reconstruction problems. Lastly, the possibility of using it in higher dimension could be applicable for manifold learning applications.

Acknowledgements This work was funded by the European Research Council (ERC Starting Grant "SurfComp") and the Israel Science Foundation. The following models are from the Aim@Shape repository: Max Planck, Bimba, Elephant, Horse, Skull, Hand, Hand2, Rocker. The Gorilla is from the Tosca repository [Bronstein et al. 2008]. The authors would like to thank Scott Schaefer for the MVC examples, and the anonymous reviewers for their useful comments and suggestions.

References

ALLIEZ, P., COHEN-STEINER, D., YVINEC, M., AND DESBRUN, M. 2005. Variational tetrahedral meshing. *ACM Trans. Graph. 24*, 3 (July), 617–625.

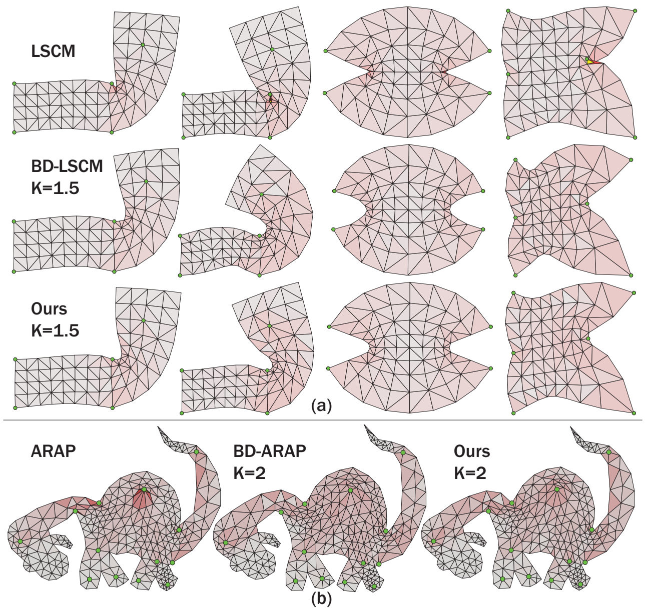


Figure 9: *Comparisons of our algorithm to results from [Lipman 2012]. In **(a)**, 4 deformations of a rectangle, generated once with LSCM ([Lévy et al. 2002]), BD-LSCM from [2012], and the projection of the LSCM map using our algorithm. In **(b)**, Deformation of a dino mesh, generated with ARAP, BD-ARAP from [2012], and the projection of the ARAP deformation using our algorithm.*

BEN-CHEN, M., WEBER, O., AND GOTSMAN, C. 2009. Variational harmonic maps for space deformation. *ACM Trans. Graph. 28*, 3 (July), 34:1–34:11.

BOTSCH, M., AND SORKINE, O. 2008. On linear variational surface deformation methods. *Visualization and Computer Graphics, IEEE Transactions on 14*, 1, 213–230.

BRONSTEIN, A., BRONSTEIN, M., AND KIMMEL, R. 2008. *Numerical Geometry of Non-Rigid Shapes*, 1 ed. Springer Publishing Company, Incorporated.

CASCÓN, J., MONTENEGRO, R., ESCOBAR, J., RODRÍGUEZ, E., AND MONTERO, G. 2009. The meccano method for automatic tetrahedral mesh generation of complex genus-zero solids. In *Proceedings of the 18th International Meshing Roundtable*. Springer, 463–480.

CGAL. Cgal, computational geometry algorithms library. http://www.cgal.org.

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.

COQUILLART, S. 1990. *Extended free-form deformation: a sculpturing tool for 3D geometric modeling*, vol. 24. ACM.

DACOROGNA, B., AND MARÉCHAL, P. 2007. Convex so (n) x so (n)-invariant functions and refinements of von neumann's inequality. In *ANNALES-FACULTE DES SCIENCES TOULOUSE MATHEMATIQUES*, vol. 16, Université Paul Sabatier, 71.

ECK, M., DEROSE, T., DUCHAMP, T., HOPPE, H., LOUNSBERY, M., AND STUETZLE, W. 1995. Multiresolution analysis of arbitrary meshes. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '95, 173–182.

EPPSTEIN, D. 2001. Global optimization of mesh quality. *Tutorial at the 10th Int*.

FLOATER, M. S., AND PHAM-TRONG, V. 2006. Convex combination maps over triangulations, tilings, and tetrahedral meshes. *Adv. Comput. Math. 25*, 4, 347–356.

FLOATER, M. S., KÓS, G., AND REIMERS, M. 2005. Mean value coordinates in 3d. *Comput. Aided Geom. Des. 22*, 7 (Oct.), 623–631.

FLOATER, M. S. 2003. One-to-one piecewise linear mappings over triangulations. *Math. Comput. 72*, 242, 685–696.

FREITAG, L. A., AND KNUPP, P. M., 2002. Tetrahedral mesh improvement via optimization of the element condition number.

GREGSON, J., SHEFFER, A., AND ZHANG, E. 2011. All-hex mesh generation via volumetric polycube deformation. *Computer Graphics Forum (Special Issue of Symposium on Geometry Processing 2011) 30*, 5, to appear.

IRVING, G., TERAN, J., AND FEDKIW, R. 2004. Invertible finite elements for robust simulation of large deformation. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '04, 131–140.

JACOBSON, A., BARAN, I., POPOVIC, J., AND SORKINE, O. 2011. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph. 30*, 4, 78.

JOHNEN, A., REMACLE, J.-F., AND GEUZAINE, C. 2012. Geometrical validity of high-order triangular finite elements. *Engineering with Computers*, 1–8.

JOSHI, P., MEYER, M., DEROSE, T., GREEN, B., AND SANOCKI, T. 2007. Harmonic coordinates for character articulation. *ACM Trans. Graph. 26*, 3 (July).

JU, T., SCHAEFER, S., AND WARREN, J. 2005. Mean value coordinates for closed triangular meshes. *ACM Trans. Graph. 24*, 3 (July), 561–566.

KLINGNER, B. M., AND SHEWCHUK, J. R. 2007. Agressive tetrahedral mesh improvement. In *Proceedings of the 16th International Meshing Roundtable*, 3–23.

LABELLE, F., AND SHEWCHUK, J. R. 2007. Isosurface stuffing: Fast tetrahedral meshes with good dihedral angles. *ACM Transactions on Graphics 26*, 3 (July), 57.1–57.10. Special issue on Proceedings of SIGGRAPH 2007.

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

LI, X., GUO, X., WANG, H., HE, Y., GU, X., AND QIN, H. 2007. Harmonic volumetric mapping for solid modeling applications. In *Proceedings of the 2007 ACM symposium on Solid and physical modeling*, ACM, New York, NY, USA, SPM '07, 109–120.

LIPMAN, Y., LEVIN, D., AND COHEN-OR, D. 2008. Green coordinates. *ACM Trans. Graph. 27*, 3 (Aug.), 78:1–78:10.

LIPMAN, Y. 2012. Bounded distortion mapping spaces for triangular meshes. *ACM Trans. Graph. 31*, 4 (July), 108:1–108:13.

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

MEYER, M., DESBRUN, M., SCHRÖDER, P., AND BARR, A. 2002. Discrete differential-geometry operators for triangulated 2-manifolds. *Visualization and mathematics 3*, 7, 34–57.

MÜLLER, M., DORSEY, J., MCMILLAN, L., JAGNOW, R., AND CUTLER, B. 2002. Stable real-time deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ACM, 49–54.

OWEN, S. 1998. A survey of unstructured mesh generation technology. In *7th International Meshing Roundtable*, vol. 3.

SEDERBERG, T., AND PARRY, S. 1986. Free-form deformation of solid geometric models. In *ACM Siggraph Computer Graphics*, vol. 20, ACM, 151–160.

SHEWCHUK, J. R. 2012. *Combinatorial Scientific Computing*. CRC Press, January, ch. Unstructured Mesh Generation, chapter 10, 257–297.

SORKINE, O., AND ALEXA, M. 2007. As-rigid-as-possible surface modeling. In *Proceedings of the fifth Eurographics symposium on Geometry processing*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SGP '07, 109–116.

STEWART, G. W. 1990. Perturbation theory for the singular value decomposition. In *IN SVD AND SIGNAL PROCESSING, II: ALGORITHMS, ANALYSIS AND APPLICATIONS*, Elsevier, 99–109.

TENG, S.-H., AND WONG, C. W. 2000. Unstructured mesh generation: Theory, practice, and perspectives. *International Journal of Computational Geometry & Applications 10*, 03, 227–266.

TOURNOIS, J., WORMSER, C., ALLIEZ, P., AND DESBRUN, M. 2009. Interleaving delaunay refinement and optimization for practical isotropic tetrahedron mesh generation. *ACM Trans. Graph. 28*, 3 (July), 75:1–75:9.

WANG, Y., GU, X., THOMPSON, P. M., AND YAU, S. T. 2004. 3d harmonic mapping and tetrahedral meshing of brain imaging data. *Proc. Medical Imaging Computing and Computer Assisted Intervention (MICCAI), St. Malo, France, Sept*.

WANG, Y., GU, X., CHAN, T. F., THOMPSON, P. M., AND YAU, S.-T. 2004. Volumetric harmonic brain mapping. In *Proceedings of the 2004 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Arlington, VA, USA, 15-18 April 2004*, IEEE, 1275–1278.

WEBER, O., MYLES, A., AND ZORIN, D. 2012. Computing extremal quasiconformal maps. *Comp. Graph. Forum 31*, 5 (Aug.), 1679–1689.

XIA, J., HE, Y., HAN, S., FU, C.-W., LUO, F., AND GU, X. 2010. Parameterization of star-shaped volumes using green&#39;s functions. In *Proceedings of the 6th international conference on Advances in Geometric Modeling and Processing*, Springer-Verlag, Berlin, Heidelberg, GMP'10, 219–235.

XU, H., YU, W., GU, S., AND LI, X. 2012. Biharmonic volumetric mapping using fundamental solutions. *IEEE Transactions on Visualization and Computer Graphics 99*, PrePrints, 1.

Appendix A

Theorem 1. *Let $A \in \mathbb{M}$ be a non-zero arbitrary matrix. The closest K-bounded-distortion matrix $\mathrm{Pr}\,(A) \in \mathbb{M}_K$ is non-zero and has the form*

$$\mathrm{Pr}\,(A) = UXV^T,$$

where $X = \mathrm{diag}(\boldsymbol{x})$ is a diagonal matrix, and $A = U\Sigma V^T$ is the SSVD of A. Furthermore,

$$\boldsymbol{x} = \mathrm{argmin}_{\mathbf{y} \,\in\, \mathcal{T}_K} \left\| \mathbf{y} - \boldsymbol{\sigma}(A) \right\|_2 ,$$

where

$$\mathcal{T}_K = \left\{ \mathbf{y} \in \mathbb{R}^3 \;\middle|\; y_1 \le K y_3 \ , \ \ y_1 \ge y_2 \ge y_3 > 0 \right\}.$$

Proof. The idea is to use the following inequality that holds for any two matrices A, B of the same dimensions,

$$\|A - B\|_F^2 \ge \sum_{\ell=1}^{3} \left(\sigma_\ell(A) - \sigma_\ell(B)\right)^2 . \qquad (12)$$

This is a generalization of the well-known Mirsky inequality [Stewart 1990] to the *signed* singular value case. We prove it in Lemma 1. Now, let $B \in \mathbb{M}_K$ be an arbitrary bounded-distortion matrix, then

$$\|A - B\|_F \overset{(12)}{\ge} \|\boldsymbol{\sigma}\,(A) - \boldsymbol{\sigma}\,(B)\|_2 \ge \|\boldsymbol{\sigma}\,(A) - \mathbf{x}\|_2 , \qquad (13)$$

where $\mathbf{x} = \mathrm{argmin}_{\mathbf{y} \,\in\, \mathcal{T}_K} \left\| \mathbf{y} - \boldsymbol{\sigma}(A) \right\|_2$ (the last inequality holds since $\boldsymbol{\sigma}\,(B) \in \mathcal{T}_K$). Let $X = \mathrm{diag}\,(\mathbf{x})$. Note $X \in \mathbb{M}_K$ since $\mathbf{x} \in \mathcal{T}_K$, and by invariance to rotation also $UXV^T \in \mathbb{M}_K$. Then

$$\left\| A - UXV^T \right\|_F = \|\Sigma - X\|_F = \|\boldsymbol{\sigma}\,(A) - \mathbf{x}\|_2 \overset{(13)}{\le} \|A - B\|_F$$

Which, since $B \in \mathbb{M}_K$ is arbitrary, shows that UXV^T is the closest bounded-distortion matrix to A. Note that since $\mathcal{T}_K$ is not closed we still need to show a minimizer exists. This is true for all A except the zero matrix, and is proved in Lemma 2.

□

Lemma 1. *For any two matrices A, B of the same dimension $\|A - B\|_F^2 \ge \sum_{\ell=1}^{3} \left(\sigma_\ell(A) - \sigma_\ell(B)\right)^2$, where $\boldsymbol{\sigma}(A), \boldsymbol{\sigma}(B)$ are the signed singular values of A, B (respectively).*

Proof. First, $\|A - B\|_F^2 = \|A\|_F^2 + \|B\|_F^2 - 2\mathrm{tr}(A^T B) = \sum_\ell \sigma_\ell(A)^2 + \sum_\ell \sigma_\ell(B)^2 - 2\mathrm{tr}(A^T B)$. And the proof is finished by using the generalized Von-Neumann inequality for signed singular values [Dacorogna and Maréchal 2007], namely, $\mathrm{tr}(A^T B) \le \sum_\ell \sigma_\ell(A)\sigma_\ell(B)$. □

Lemma 2. *There exists a non-zero minimizer to the problem $\boldsymbol{x} = \mathrm{argmin}_{\mathbf{y} \,\in\, \mathcal{T}_K} \left\| \mathbf{y} - \boldsymbol{\sigma}(A) \right\|_2$, for any non-zero matrix A.*

Proof. It is enough to show that for a non-zero $\mathbf{x}$ the point $\mathbf{0} = (0, 0, 0)$ cannot achieve the minimal distance. Remember that $\sigma_1(A) \ge \sigma_2(A) \ge |\sigma_3(A)| > 0$. Consider the function $g(t) = \sum_{\ell=1}^{3} (t - \sigma_\ell(A))^2$, and note that $g'(0) < 0$. This implies that taking $\mathbf{y} = (t, t, t)$ for sufficiently small $t > 0$ will satisfy $\|\mathbf{y} - \boldsymbol{\sigma}(A)\|_2 < \|\mathbf{0} - \boldsymbol{\sigma}(A)\|_2$. □

Theorem 2. *$\mathbb{M}_K^*(A)$ is a good approximation to $\mathbb{M}_K$ in the sense that for (almost) all $B \in \mathbb{M}_K^*(A)$, $\mathrm{dist}\,(B, \mathbb{M}_K) = \mathcal{O}(\|E\|_F^2)$, where E is defined in eq. (11). That is, the distance from B to $\mathbb{M}_K$ is of order of the* squared *norm of E.*

Proof. It is enough to show the theorem for $B = D + E \in \mathcal{D}_K^*$. Indeed, for $B' \in \mathbb{M}_K^*(A) = U\mathcal{D}_K^* V^T$, $B' = UBV^T$, and $B \in \mathcal{D}_K^*$, we have $\mathrm{dist}\,(B', \mathbb{M}_K) = \|B' - \mathrm{Pr}\,(B')\,\|_F \le \|UBV^T - U\mathrm{Pr}\,(B)\,V^T\|_F = \|B - \mathrm{Pr}\,(B)\,\|_F = \mathrm{dist}\,(B, \mathbb{M}_K) = \mathcal{O}(\|E\|_F^2)$, where in the inequality we used the fact that $U\mathrm{Pr}\,(B)\,V^T \in \mathbb{M}_K$.

The most direct way to show this property is using perturbation theory for singular values [Stewart 1990]. let $B = D + E \in \mathcal{D}_K^*$, where $D \in \mathcal{D}_K$ and $E \in \mathcal{E}$. According to the perturbation expansion presented in Section 4 in [1990], For a simple singular value $\sigma_i(D)$ we have

$$\sigma_i(B) = \sigma_i(D) + U_i^T UEV^T V_i + \mathcal{O}\left(\left\| UEV^T \right\|_F^2 \right),$$

where U_i, V_i are the i^{th} column vectors of U, V (respectively). Note that $U_i^T U = \mathbf{e}_i^T$, $V^T V_i = \mathbf{e}_i$, where $\mathbf{e}_i \in \mathbb{R}^{3\times1}$ is the standard vector of all zeros with the exception of the i^{th} entry that equals one. Since E has zeros on its diagonal, $U_i^T UEV^T V_i = \mathbf{e}_i^T E \mathbf{e}_i = 0$. Using that and the fact that the Frobenius norm is invariant to left and right multiplications by orthogonal matrices we get

$$\sigma_i(B) = \sigma_i(D) + \mathcal{O}\left(\|E\|_F^2\right).$$

The Frobenius distance of the matrix B to $\mathbb{M}_K$ can now be bounded from above,

$$\begin{aligned} \mathrm{dist}\,(B, \mathbb{M}_K) &= \min_{\mathbf{y} \in \mathcal{T}_K} \|\mathbf{y} - \boldsymbol{\sigma}(B)\|_2 \\ &\le \|\boldsymbol{\sigma}(B) - \boldsymbol{\sigma}(D)\|_2 \\ &= \mathcal{O}(\|E\|_F^2), \end{aligned}$$

where in the inequality we used the fact that $\boldsymbol{\sigma}(D) \in \mathcal{T}_K$.

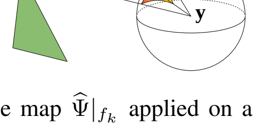
One comment is that we assumed in the above analysis that $\sigma_i(D)$ is simple. This assumption breaks whenever two entries in the diagonal of D are equal. Note that these cases correspond to the edges in the cone $\mathcal{D}_K$, and indeed the approximation result holds away from the edges. □

Appendix B

We prove bijectivity of tetrahedral simplicial maps:

Theorem 3 *A simplicial map $\Psi \in \mathcal{F}$ that maps the boundary of $\boldsymbol{M}$ bijectively onto the boundary of domain Ω, and that the differentials B_j of its affine maps $\Psi|_{t_j}$ satisfy eq. (3), that is $\sigma_3(B_j) > 0$ (or equivalently $\det B_j > 0$) is a global bijection between $\boldsymbol{M}$ and Ω.*

Proof. Let us fix an arbitrary point $\mathbf{y} \in \mathbb{R}^3 \setminus \cup_j \Psi(\partial t_k)$, that is a point that is not in the image of any face, edge of a vertex of $\mathbf{M}$. Then our first task is to count the number of pre-images $\#\left\{\Psi^{-1}(\mathbf{y})\right\}$. For that end we will need a little preparation. We will abuse notation and denote by $\mathbf{M}$ also the set of points (all points, not just vertices) consisting of the tet mesh. We denote $\mathbb{S}(\mathbf{y})$ the unit sphere centered at $\mathbf{y}$. Let $\mathbf{S} \subset \mathbf{M}$ be a closed polyhedral surface consisting of vertices, edges, and faces of the tet mesh $\mathbf{M}$. We denote by $\chi : \mathbb{R}^3 \setminus \{\mathbf{y}\} \to \mathbb{S}(\mathbf{y})$ the map $\chi(\mathbf{x}) = \mathbf{y} + \frac{\mathbf{x}-\mathbf{y}}{\|\mathbf{x}-\mathbf{y}\|}$, mapping all points in $\mathbb{R}^3$ except $\mathbf{y}$ onto the unit sphere around $\mathbf{y}$. We would like to use the notion of *degree* of the map $\widehat{\Psi} = \chi \circ \Psi$ restricted to a polyhedral surface $\mathbf{S}$, that is $\deg \widehat{\Psi}|_{\mathbf{S}}$. The inset shows the map $\widehat{\Psi}|_{f_k}$ applied on a



single face f_k. The *degree* of a map is a generalization of a winding number to higher dimensions and intuitively counts how many times $\mathbf{S}$ covers $\mathbb{S}(\mathbf{y})$ under the map $\widehat{\Psi}$. The number $\deg \widehat{\Psi}|_{\mathbf{S}}$ can be computed by the following steps: 1) choose a point $\mathbf{x} \in \mathbb{S}(\mathbf{y})$ that is not an image of an edge or a vertex of $\mathbf{S}$; 2) take all faces $f_1, .., f_L$ of $\mathbf{S}$ that contain a pre-image $\mathbf{x}$; 3) then $\deg \widehat{\Psi}|_{\mathbf{S}} = \sum_{\ell=1}^{L} \tau_\ell$, where $\tau_\ell = 1$ if the orientation of f_ℓ is kept under the map $\widehat{\Psi}$ w.r.t. the orientations of $\mathbf{S}$ and $\mathbb{S}(\mathbf{y})$, and $\tau_\ell = -1$ otherwise.

A closed oriented polyhedral surface $\mathbf{S}$ can be represented as a *cycle*, that is a formal sum of its faces, where each coefficient is +1 if the orientation of the surface coincides with the orientation of the corresponding face, and -1 otherwise. We will not distinguish between surfaces and their cycles from now on. Now let $\mathbf{S}_1, \mathbf{S}_2$ be two (two-dimensional) cycles of $\mathbf{M}$. Their sum $\mathbf{S} = \mathbf{S}_1 + \mathbf{S}_2$ is also a cycle (closed oriented polyhedral surface). The degree has the following property

$$\deg \widehat{\Psi}|_{\mathbf{S}_1+\mathbf{S}_2} = \deg \widehat{\Psi}|_{\mathbf{S}_1} + \deg \widehat{\Psi}|_{\mathbf{S}_2}, \tag{14}$$

as can be checked directly from its definition.

Now we can count the number of pre-images of the point $\mathbf{y}$. Since the matrices B_j of the affine maps $\Psi|_{t_j}$ are orientation-preserving,

$$\deg \widehat{\Psi}|_{\partial t_j} = \begin{cases} 1 & \mathbf{y} \in \Psi(t_j) \\ 0 & \mathbf{y} \notin \Psi(t_j) \end{cases},$$

where ∂t_j is the boundary polyhedral surface of the tet t_j. Therefore,

$$\#\left\{\Psi^{-1}(\mathbf{y})\right\} = \sum_{t_j \in \mathbf{T}} \deg \widehat{\Psi}|_{t_j} = \deg \widehat{\Psi}|_{\partial \mathbf{M}},$$

where in the second equality we used eq. (14). Finally, we are left with the relatively easy task of calculating $\deg \widehat{\Psi}|_{\partial \mathbf{M}}$: since $\Psi|_{\partial \mathbf{M}}$ is a bijection onto $\partial\Omega$, it is enough to compute $\deg \chi|_{\partial\Omega}$:

$$\deg \chi|_{\partial\Omega} = \begin{cases} 1 & \mathbf{y} \in \Omega \\ 0 & \mathbf{y} \notin \Omega \end{cases} \tag{15}$$

So far we proved that: 1) Ψ is injective when restricted to the set $\Upsilon = \{\mathbf{x} | \mathbf{x} \in \mathbf{M},\ \Psi(\mathbf{x}) \notin \cup_j \Psi(\partial t_k)\}$, i.e. the set of all points in $\mathbf{M}$ that their image is not on the image of any face, edge or vertex; 2) $\Psi(\Upsilon) \subset \Omega$; and 3) $\text{Closure}(\Psi(\Upsilon)) = \Omega$. To finish the proof we will use Lemma 3 that indicates that Ψ is an *open map*, namely mapping open sets to open sets. Once we know Ψ is an open map we can show both that $\Psi(\mathbf{M}) = \text{Closure}(\Omega)$, and that Ψ is injective over $\mathbf{M}$. First, assume there is a point $\mathbf{x} \in \text{Interior}(\mathbf{M})$ such that $\Psi(\mathbf{x}) \notin \text{Interior}(\Omega)$. Then since Ψ is open it maps a small open neighborhood $U \subset \mathbf{M}$ of $\mathbf{x}$ to an open neighborhood of a point $\mathbf{z} \notin \text{Interior}(\Omega)$. This means there has to exist some point $\mathbf{z}' \in \Psi(\Upsilon) \setminus \text{Closure}(\Omega)$, that is, there is a point in $\Psi(\Upsilon)$ outside Ω, in contradiction to claim 2. This shows $\Psi(\text{Interior}(\mathbf{M})) \subset \text{Interior}(\Omega)$. On the other hand take any point $\mathbf{y} \in \text{Interior}(\Omega)$, it is in the closure of $\Psi(\Upsilon)$. Therefore by continuity of Ψ and compactness of $\mathbf{M}$ we have that $\mathbf{y} \in \Psi(\mathbf{M})$. So we get $\Psi(\text{Interior}(\mathbf{M})) = \text{Interior}(\Omega)$. Since we know $\Psi(\partial\mathbf{M}) = \partial\Omega$,we have $\Psi(\mathbf{M}) = \Omega$. Injectivity is shown by assuming $\mathbf{z}_1 \neq \mathbf{z}_2 \in \mathbf{M}$ such that $\Psi(\mathbf{z}_1) = \Psi(\mathbf{z}_2)$. We can assume $\mathbf{z}_1, \mathbf{z}_2 \in \text{Interior}(\mathbf{M})$, since the boundary is mapped bijectively and we saw $\Psi(\text{Interior}(\mathbf{M})) = \text{Interior}(\Omega)$. Since Ψ is an open map if $\Psi(\mathbf{z}_1) \notin \Psi(\Upsilon)$ we can perturb it by an arbitrary small amount and find two new points $\mathbf{z}_1', \mathbf{z}_2' \in \text{Interior}(\mathbf{M})$ such that $\Psi(\mathbf{z}_1') = \Psi(\mathbf{z}_2') \in \Psi(\Upsilon)$, in contradiction to Claim 1. □

Lemma 3. *A simplicial map $\Psi \in \mathcal{F}$ such that its differential matrices B_j satisfy eq. (3), is an open map.*

Proof. It is enough to show that for every interior point $\mathbf{p} \in \text{Interior}(\mathbf{M})$ there exists a small open ball $B_{\mathbf{p}}$ (that can be made arbitrarily small) centered at $\mathbf{p}$, such that $\Psi(\mathbf{p}) \in \text{Interior}(\Psi(B_{\mathbf{p}}))$. That is, $\Psi(\mathbf{p})$ is interior point of the set $\Psi(B_{\mathbf{p}})$.

We will have four different cases: 1) $\mathbf{p}$ is interior to a tet, 2) $\mathbf{p}$ is interior to a face, 3) $\mathbf{p}$ is interior to an edge, and 4) $\mathbf{p} = \boldsymbol{v}_i$ is a vertex.

All cases can be explained by using the degree argument as follows. Case 1 is actually even simpler as it directly follows from the fact that since the differentials are not degenerate, the interior of each tet is mapped homeomorphically. For the other cases we take the two dimensional cycle $\mathbf{S}$ defined as the boundary of the union of all tets adjacent to $\mathbf{p}$. We define the map χ similar to above by taking $\mathbf{y} = \Psi(\mathbf{p})$, and denoting $\widehat{\Psi} = \chi \circ \Psi$. Now, by definition $\deg \widehat{\Psi}|_{\mathbf{S}}$ is an integer. Since all tets have to preserve their orientation by assumption, this integer has to be at-least one. Indeed, we can take a point $\mathbf{z} \in \mathbb{S}(\Psi(\mathbf{p}))$ that has at-least one pre-image inside one of the faces of $\mathbf{S}$. Since the faces are not inverted this pre-image will contribute $+1$ to the degree. As no faces of $\mathbf{S}$ are inverted the degree cannot be smaller than 1. In particular this means that every point $\mathbf{x} \in \mathbb{S}(\Psi(\mathbf{p}))$ has a pre-image of $\widehat{\Psi}$ in $\mathbf{S}$. Hence, every point in a sufficiently small ball around $\mathbf{y} = \Psi(\mathbf{p})$ is in $\Psi(B_{\mathbf{p}})$.

□