

Linear Surface Reconstruction from Discrete Fundamental Forms on Triangle Meshes

Y. Wang B. Liu Y. Tong

Michigan State University
 wangyua6@msu.edu liubeibe@msu.edu ytong@msu.edu

Abstract

We present a linear algorithm to reconstruct the vertex coordinates for a surface mesh given its edge lengths and dihedral angles, unique up to rotation and translation. A local integrability condition for the existence of an immersion of the mesh in 3D Euclidean space is provided, mirroring the fundamental theorem of surfaces in the continuous setting (i.e., Gauss's equation and the Mainardi-Codazzi equations) if we regard edge lengths as the discrete first fundamental form and dihedral angles as the discrete second fundamental form. The resulting sparse linear system to solve for the immersion is derived from the convex optimization of a quadratic energy based on a lift from the immersion in the 3D Euclidean space to the 6D rigid motion space. This discrete representation and linear reconstruction can benefit a wide range of geometry processing tasks such as surface deformation and shape analysis. A rotation-invariant surface deformation through point and orientation constraints is demonstrated as well.

Keywords: differential coordinates, first fundamental forms, second fundamental forms, surface deformation

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

Local rigid motion invariant representations often serve as a starting point for differential geometry of surfaces. In the discrete setting, such representations are also of great interest in various geometry processing contexts. For geometric modeling, local encoding of geometric details facilitates intuitive surface manipulation and deformation. For the simulation of elastic thin-shell objects, the strain energy must only depend on invariants of rigid motions to ensure preservation of momenta. Even parameterization, geometric texture synthesis, shape analysis, or 3D model registration can potentially benefit from an efficient rotation-invariant description of the shape.

An obvious choice for such a representation for surface meshes would be to use edge lengths and dihedral angles, which uses approximately twice as much storage space as the original vertex coordinates, but is indeed rotation-invariant and local. Moreover, given an arbitrary parameterization, they can be used to construct the first and sec-

ond fundamental forms, which describe the metric of the surface and the bending of normals respectively. However, constructing a surface embedding (or immersion) in a linear fashion from an *arbitrary* set of edge lengths and dihedral angles is far from straightforward—except when the set of values are known to come from an existing embedding, in which case a sequential reconstruction starting from a seed vertex is trivial.

In this paper we provide a (necessary and sufficient) local integrability condition for a given triangle connectivity to admit an immersion in \mathbb{R}^3 for a set of prescribed edge lengths and dihedral angles. We also present a linear approach to determining an embedding even when there are conflicting point and orientation constraints, or when the edge lengths and dihedral angles fail to satisfy our integrability conditions.

Related Work With the increase in computing power and user demands, mesh deformation and other related geometry processing methods have become popular tools. Their

goal is often to produce reasonable, natural-looking surfaces that satisfy user-specified deformation constraints at near interactive rates. There are mainly two types of algorithms: surface-based, and ambient-space-based (also called free-form) deformation methods. As our contribution fits squarely in the first category, we mainly go over related surface-based methods. Some among these methods focus on point sampled geometry in a global coordinate system and employ a scalar (blending) field to get a smooth transition between constrained and unconstrained parts. For instance, [BK03] uses geodesic distances to define the blending function and adds an implicit occluder function to the mesh editing environment, while [PKKG03] uses Euclidean distances in a hybrid geometry model combining implicit and parametric surface representations. The latter produces more intuitive deformation at the cost of increased computational time. A number of authors, seeking physically plausible results, propose to minimize various physically-motivated deformation energies. For example, [CPSS10] defines an elastic energy based on the distance between the deformation and the rotation group, including an additional thin-shell bending term if necessary; [WBH^{*}07] provides instead a quadratic curvature-based energy, thus resulting in simpler computations. To preserve geometric details for fine-scale surfaces, other authors suggested to combine physically-motivated energy methods with multi-resolution modeling [KCVS98, KVS99, BK04]. Alternatively, several papers [SCOL^{*}04, LSCO^{*}04, YZX^{*}04, ZHS^{*}05, ZRKS05] argue that geometric detail is a local intrinsic property of the surface and propose to use so-called *differential coordinates* to encode these details instead of storing them in a global coordinate system. For instance, [YZX^{*}04, ZRKS05] define differential coordinates based on the gradients of original surface coordinates, while [ZHS^{*}05, LSCO^{*}04, SCOL^{*}04] manipulate the Laplacian of the spatial coordinates. See [BS08] for a survey on these methods.

The deformation technique that we derive from our discrete surface representation can be seen as a linear method within differential-coordinates methods. Among those, [LSCO05] is the only existing linear and rotation-invariant approach (not including iterative methods with linear steps, of course). It defines a local frame per vertex and encodes the local geometric information into the transition between adjacent local frames. The method produces deformations that preserve local details in the least-squares sense through a linear solve for the local frames and another for the coordinates. It handles rotational constraints efficiently, but is insensitive to translational constraints, since the solve for the frames is decoupled from the positional constraints [BS08]. [LCOGL07] developed a mesh editing method that aims at preserving shape and volume based on this representation. Our method strives to achieve the same goal, but in *sharp contrast*, we provide a natural and compact description of the fundamental forms that does not rely on the existence of an original embedding, the choice of a

local vertex normal, or the choice of picking an edge as the first tangential direction, and produces mesh deformation in a single linear system. [KG08] also proposed to attach one local frame for each triangle face, but the frames are not orthogonal and the linear system cannot be derived from given edge lengths and dihedral angles without an existing mesh. Edge lengths and dihedral angles have been used as the local shape descriptor in the physics-based *nonlinear* energy in [FB11] to generate natural looking mesh deformation. They are also used for geometry interpolation in [WDAH10], where computation of their interpolation before the vertex location interpolation was proposed to better preserve local shape.

Contributions

Our main contributions include:

- A local and a global version of the discrete fundamental theorem of surfaces, based on the first and second fundamental forms represented simply by edge lengths and dihedral angles;
- A simple procedure to construct an embedding through a sparse linear solve given an arbitrary set of edge lengths and dihedral angles;
- A practical mesh deformation algorithm for handling point constraints and orientation constraints concurrently.

The rest of the paper is organized as follows. We first provide important continuous definitions in Section 2.1, before proving in Section 2.3 a discrete fundamental theorem of surfaces, through a lift from the 3D Euclidean immersion to a 6D immersion (described in Section 2.2). We then present an energy function, the minimization of which gives an optimal immersion for a given set of edge lengths and dihedral angles in Section 3. We describe the discretization and implementation details in Section 4. Deformation results are shown in Section 5. We conclude and discuss future work in Section 6.

2. A Discrete Fundamental Theorem of Surfaces

2.1. Background

We briefly go through the concepts involved in the continuous fundamental theorem of surfaces, roughly following the exposition of [Fra03] (Sec. 8.5). For a more detailed treatment on the subject see, e.g., [IL03] (Chapter 2).

For a surface M with a parameterization (u^1, u^2) , if there is a local embedding \mathbf{x} in \mathbb{R}^3 , a first fundamental form (metric tensor) I can be induced by $I_{\alpha\beta} = \langle \mathbf{x}_\alpha, \mathbf{x}_\beta \rangle$, and a second fundamental form II by $II_{\alpha\beta} = \langle \mathbf{x}_{,\alpha\beta}, \mathbf{N} \rangle$, where \mathbf{N} denotes the normal, lowercase Greek letters are used for indices $\in \{1, 2\}$, and an index after a comma in the subscript denotes partial derivatives. Intuitively, I measures lengths on the surface induced by the 3D Euclidean metric, and II measures the rate of change of the normal field. We can use I or its inverse to raise or lower indices for tensors defined on

the surface. We use Einstein's summation convention in the following discussion.

Gauss's surface equations express the derivatives of the tangent vectors $\mathbf{x}_{,\alpha}$, in the local frame formed by $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{N}\}$ as

$$\mathbf{x}_{,\alpha\beta} = \Gamma_{\alpha\beta}^\gamma \mathbf{x}_{,\gamma} + II_{\alpha\beta} \mathbf{N}$$

where $\Gamma_{\alpha\beta}^\gamma$ is the Christoffel symbol of I . Note that if we treat the vector $\mathbf{x}_{,\alpha\beta}$ as a linear combination of the basis vectors in the local frame, all the coefficients (i.e., $\Gamma_{\alpha\beta}^\gamma$ and $II_{\alpha\beta}$) in the above equations can be derived from I and II . Roughly speaking, given a symmetric positive definite matrix function I and a symmetric matrix function II , if we start from a seed point and a chosen pair of tangential directions $\mathbf{x}_{,1}$ and $\mathbf{x}_{,2}$, we can propagate these directions along any paths on the surface by integrating these equations. We can then construct an immersion by integrating $\mathbf{x}_{,1}$ and $\mathbf{x}_{,2}$.

I and II provide 6 numbers per point while the 3D immersion requires only 3 numbers (coordinates) per point, so there must be 3 equations per point that I and II must satisfy. In fact, as the left-hand side of the above equation describes the gradient of $\mathbf{x}_{,\alpha}$, it has to be curl-free, which means

$$0 = \mathbf{x}_{,\alpha\beta\gamma} - \mathbf{x}_{,\alpha\gamma\beta} = (R_{\alpha\beta\gamma}^\tau - (II_\gamma^\tau II_{\alpha\beta} - II_\beta^\tau II_{\alpha\gamma}))\mathbf{x}_{,\tau} + (\nabla_\gamma II_{\alpha\beta} - \nabla_\beta II_{\alpha\gamma})\mathbf{N}, \quad (1)$$

where R denotes the Riemann curvature tensor determined by I , and ∇ denotes the covariant derivative defined by I . The vanishing of the coefficients in front of $\mathbf{x}_{,\tau}$ (resp., \mathbf{N}) is called *Gauss's equation* (resp., the *Mainardi-Codazzi equations*). Note that on surfaces, $R_{\alpha\beta\gamma}^\tau$ has only one independent component R^1_{212} ; thus Gauss's equation simply states that the Gaussian curvature $\det(I^{-1}II)$ is the intrinsic curvature.

The fundamental theorem of surfaces states that Gauss's equation and the Codazzi equations are not only a necessary condition for I and II to be induced by a local embedding, but also a sufficient one, as proven using the Frobenius theorem; see e.g. [IL03]. Intuitively speaking, the Codazzi equations guarantee that the propagation of the normal direction (determined by the two tangential directions) from a seed point is independent of the choice of paths one takes, while Gauss's equation guarantees that the rotation in the tangent plane (defined by the two tangent directions) is consistent with the parallel transport prescribed by the metric I , and thus also path-independent. The equations are seen as *integrability conditions*, as we may regard the surface embedding as the integral of I and II , which are the rigid motion invariants of the first and second derivatives of the embedding. We will prove our discrete version of the fundamental theorem of surfaces based on these observations next.

2.2. Lift of an Immersion

In our proof of the discrete fundamental theorem of surfaces

and in our derivation of an efficient algorithm, we employ a convenient mathematical tool called lift, which is also used in the above-mentioned proof of the smooth case. The lift $F(u^1, u^2)$ of an immersion $\mathbf{x}(u^1, u^2)$ from the 3D Euclidean space to the 6D rigid transformation (or Euclidean motion) group $E(3)$ is expressed in the familiar 4×4 -matrix form for homogeneous coordinates

$$F = \begin{pmatrix} f & \mathbf{x} \\ 0 & 1 \end{pmatrix},$$

where $f = (\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3)$ is a rotation matrix representing a local frame at \mathbf{x} . We further restrict the third basis vector \mathbf{b}_3 to be \mathbf{N} , and thus $(\mathbf{b}_1, \mathbf{b}_2)$ forms an orthonormal frame of the tangent plane. This moving frame that we attach to each point on surfaces is similar to the Frenet-Serret frame for curves. dF provides the differential properties of the surface, including the tangential directions ($d\mathbf{x}$) and how the frames are changing (df). To find the differential invariants, we describe these quantities in the frame with the axes aligned to f and the origin located at \mathbf{x} , i.e., we use $F^{-1}dF$. Technically, this turns out to be the pullback of a canonical Lie-algebra-valued 1-form on $E(3)$, the Maurer-Cartan form Ω . Roughly speaking, it just identifies the tangent space at any F to the tangent space at the identity of the group through the group action (i.e., matrix multiplication) of F^{-1} ; refer to [IL03] for the mathematical definitions. In this context, the pullback of Ω onto the surface simply means that we restrict the application of the 1-forms to only tangent vectors of the surface. We omit the pullback notation below for simplicity.

$$\begin{aligned} \Omega &= F^{-1}dF = \begin{pmatrix} f^T & -\mathbf{x} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} df & d\mathbf{x} \\ 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} f^T df & f^T d\mathbf{x} \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\omega_1^2 & -\omega_1^3 & \omega^1 \\ \omega_1^2 & 0 & -\omega_2^3 & \omega^2 \\ \omega_1^3 & \omega_2^3 & 0 & \omega^3 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \end{aligned}$$

which can be seen as a matrix with each entry being a 1-form. $\Omega_x = (\omega^1, \omega^2, \omega^3)^T$ denotes the change of \mathbf{x} in the local frame. $\omega^3 = 0$ on the surface, since an infinitesimal move on the surface does not contain any normal component. ω^1 and ω^2 can be used as an orthonormal basis for 1-forms on the surface, thus they provide information about I ; more precisely, $I = \omega^1 \otimes \omega^1 + \omega^2 \otimes \omega^2$. The upper left 3×3 -block Ω_R denotes the rotation of f under infinitesimal moves, and is antisymmetric because f is orthonormal. ω_1^2 is exactly the metric connection form of I in the given orthonormal local frame field. We also have $\omega_\alpha^3 = II_{\alpha\beta}\omega^\beta$, describing the rotation of the normal vector. Thus, Ω encapsulates all the information in I and II , and prescribes the differential dF . Since $dF = F\Omega$,

$$\begin{aligned} d\mathbf{x} &= f\Omega_x = \omega^1\mathbf{b}_1 + \omega^2\mathbf{b}_2 \\ df &= f\Omega_R = f \begin{pmatrix} 0 & -\omega_1^2 & -\omega_1^3 \\ \omega_1^2 & 0 & -\omega_2^3 \\ \omega_1^3 & \omega_2^3 & 0 \end{pmatrix}. \end{aligned}$$

The main difference between these equations and Gauss's surface equations is that here we choose an orthonormal frame field instead of the frame induced by the parameterization. Therefore, up to a change of frames, the integrability conditions are exactly the same as Gauss's equation and the Codazzi equations. However, by introducing the intermediate variables f , the integrability conditions in this form can be constructed straightforwardly. Briefly speaking, our discrete fundamental theorem of surface examines the restrictions on the values Ω can assume to make dF integrable, and our discrete reconstruction deals with the problem of finding a least squares solution minimizing the difference between $F^{-1}dF$ and the prescribed Ω , when the system is not exactly integrable.

2.3. Discretization

We now mirror the continuous exposition presented above, but for the case of triangulated surfaces. The idea of constructing a lifted immersion in the discrete setting has been proposed before, e.g., in [LSLCO05]. However, our discretization departs sharply from this previous method in that we store *frames f on faces* (as advocated in [CDS10]) while keeping the *positions x on vertices*. Thus the rotation component and translation component of the lift are stored on *different* parts of the mesh, unlike the continuous case where F is associated to each point on the surface. With this adjustment, we do not need to specify a normal direction per vertex, and can strictly enforce $\omega^3 = 0$ in the discrete version. Moreover, this choice significantly simplifies the discrete representation of discrete fundamental forms, which can produce all the coefficients required in the discrete surface equations.

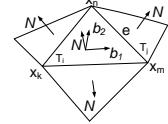
Discrete fundamental forms Given a fixed connectivity of the surface mesh M with vertex set V , edge set E , and triangle set T (which corresponds to fixed topology in the continuous case), we define the discrete first fundamental form I to be the set of edge lengths $\{l_e\}$ and the discrete second fundamental form II to be the set of angles between face normals $\{\phi_e\}$ (or ϕ_{ij} , if e is adjacent to triangles i and j). We will also sometimes refer to ϕ_e as the dihedral angle, although technically, they are π minus each other. Both forms are stored on edges; they are naturally invariant under rigid-body transformations, and independent of the parameterization.

Given an arbitrary parameterization, edge lengths can certainly be used to construct a piece-wise constant first fundamental form matrix. The dihedral angles are often used to compute mean curvatures at edges locally. However, together with the connectivity of the mesh, they provide more information than just the mean curvature, since they denote the change of the normal across the edges. In fact, given the change of normal in the three directions, one can construct a second fundamental form matrix (represented by 3 numbers in any given parameterization) per face. Conversely, the edge

lengths and dihedral angles can be computed from the fundamental form matrices. Note that no existing embedding is required in the process.

Local frames Given the discrete first fundamental forms I , we can define a 2D local orthonormal frame $(\mathbf{t}, \mathbf{t}^\perp)$ in each triangle T . This is uniquely determined by the edges and an arbitrary rotation from, e.g. the first edge of the triangle. We show later that all our derivations and algorithms are independent of this rotation. Each edge (seen as a 2D vector) can then be expressed as a linear combination of its local frame as $\mathbf{x}_n - \mathbf{x}_m = a_{mn,T}^1 \mathbf{t} + a_{mn,T}^2 \mathbf{t}^\perp$, if the edge points from vertex m to vertex n , and \mathbf{x}_m is the 2D coordinates of vertex m . Here, $a_{mn,T}^\alpha$'s are literally specifying a piecewise constant ω^1 and ω^2 given the choice of the 2D frame.

If there is an immersion, we can identify the 2D frame with $(\mathbf{b}_1, \mathbf{b}_2)$, and the inclusion map indicates $\mathbf{x}_n - \mathbf{x}_m = a_{mn,T}^1 \mathbf{b}_1 + a_{mn,T}^2 \mathbf{b}_2$ (see inset).



Transition rotation In the continuous setting, a proper integration of the rotation part of $dF = F\Omega$ along a dual path from T_i to T_j would produce $f_j - f_i = f_i R_{ij} - f_i$, where R_{ij} is a rotation determined by Ω independent of the path if Ω comes from an integrable pair of first and second fundamental form. In the discrete setting, we can use the *Transition Rotation* R_{ij} defined on each edge to express the rotation part of the discrete Ω as $R_{ij} - Id$. A discrete transition rotation that is completely determined by I and II (and consistent with an immersion when it exists) can be constructed as follows. A rotation angle that would align the first axis in the 2D orthonormal frame in a triangle T to one of its edge e is denoted as $\theta_{T \rightarrow e}$. We define R_{ij} to be the 3D rotation matrix $R_z(\theta_{T_i \rightarrow e}) R_x(\phi_e) R_z(-\theta_{T_j \rightarrow e})$, where $R_u(\theta)$ represents the rotation of axis u and angle θ , e is the shared edge with triangle T_i to its left side, and ϕ_e changes sign if the edge orientation is reversed. Basically we align the first frame axis of T_i to the common edge, rotate around the edge by the dihedral angle, and then rotate within the next face to fit the local frame in T_j .

Given an immersion with a lift F , we also have $R_{ij} = f_i^{-1} f_j$. $\theta_{T \rightarrow e}$ determines the discrete connection for tangential fields [CDS10], corresponding to ω_1^2 in the continuous setting, while ϕ_e corresponds to ω_1^3 and ω_2^3 .

Discrete surface equation Since we put f and \mathbf{x} at different locations, we need to process the two components of dF at appropriate locations.

Frame Equations The frames of the lift must satisfy

$$\forall e_{ij} \in E, \quad f_j - f_i - f_i(R_{ij} - Id) = f_j - f_i R_{ij} = 0 \quad (2)$$

Edge Equations For any triangle T containing vertex m

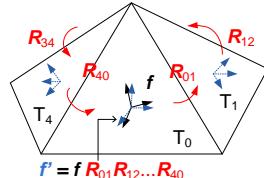
and vertex n

$$\mathbf{x}_n - \mathbf{x}_m - (a_{mn,T}^1 \mathbf{b}_{1,T} + a_{mn,T}^2 \mathbf{b}_{2,T}) = 0. \quad (3)$$

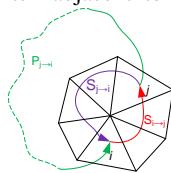
Local discrete fundamental theorem of surfaces Given a surface mesh with fixed connectivity and disk-like topology, if the transition rotation matrices $\{R_{ij}\}$ computed from a set of edge lengths (satisfying the triangle inequalities) and a set of dihedral angles satisfy the following condition for a counterclockwise loop of n triangles T_i around each interior vertex

$$\Pi_{i \in 0, \dots, n-1} R_{i, (i+1) \bmod n} - Id = 0, \quad (4)$$

then there exists an immersion in 3D Euclidean space with these edge lengths and dihedral angles as its first and second fundamental forms.



Proof The condition (Eq. 4) means that f can be updated by repeatedly applying transition rotations ($f_j = f_i R_{ij}$) along any minimal dual cycle (formed by the dual edges of the one-ring of a vertex, see inset above), and it will get back to the same frame. We prove next that integrating f using Eq. 2 along two arbitrary dual paths from face i to face j will produce the same frame f_j from the same f_i . Without loss of generality, we restrict the paths to be non-intersecting and without self-intersection. First notice that due to the trivial topology we assume for now, such two paths form a counterclockwise boundary loop around a simply connected region if we reverse one of the paths. Second, we use induction. If a dual loop encloses a single interior vertex, the accumulated rotation around it will be identity. Assume that the conclusion is true for a dual loop enclosing k vertices. For a dual loop enclosing $k+1$ vertices, we can always find one vertex adjacent to the dual loop such that removing it will leave the remaining region simply connected. Denote the common segment of the dual loop around the removed vertex and the original loop by $S_{i \rightarrow j}$, the other part of the original loop by $P_{j \rightarrow i}$, and the other part of the one-ring dual loop by $S_{j \rightarrow i}$ (See above inset figure). We have $R_{P_{j \rightarrow i}} R_{S_{i \rightarrow j}} = (R_{P_{j \rightarrow i}} R_{S_{j \rightarrow i}}^{-1})(R_{S_{j \rightarrow i}} R_{S_{i \rightarrow j}}) = Id$, as the product inside the first set of parentheses is equal to identity by the induction hypothesis.



We thus proved that we can compute all the f_i 's following a dual spanning tree from a seed face, and they are consistent with the given data. Now we prove that we can also unambiguously reconstruct the vertex locations starting from a seed vertex. First, notice that with f_j consistent with R_{ij} , the edge vector (in Eq. 3) is the same when we use the edge equation from either face. Second, by construction, $a_{ij}^\alpha + a_{jk}^\alpha + a_{ki}^\alpha = 0$, which means moving along each small

primal edge loop around each face will create no translation. We can immediately see that the coordinates \mathbf{x} can be integrated consistently since translations are commutative, unlike rotations. \square

Discussion Note that we do not need an original immersion in order to locally check the integrability. Also, the conditions are not dependent on the choice of the 2D frames, since the rotation will affect both $R_z(-\theta_{T \rightarrow e})$ and $R_z(\theta_{T \rightarrow e'})$, canceling the effect in the product. This single condition actually contains 3 independent integrability conditions as 3D rotation has 3 DoFs. We can interpret the condition as the requirement for a normal to be propagated back to itself (2 DoFs), and an additional rotation in the tangent place (1 DoF). The former corresponds to the two Codazzi equations on the symmetry of the covariant derivative of H ; the latter corresponds to Gauss's equation stating that the Gauss curvature computed by $\theta_{T \rightarrow e}$ should be consistent with the area on the unit sphere formed by the N_i 's along the dual path.

Global discrete fundamental theorem of surfaces Given a closed mesh with genus g , if, for any one homology basis, (i) the above local integrability condition is satisfied on all but two arbitrarily chosen adjacent vertices, (ii) the accumulated rotation along each dual homology generator (nontrivial loop) is the identity, and (iii) the accumulated translation (following the rotated frames) along each primal homology generator is 0, we can create an immersion of the surface mesh in 3D, unique up to a rigid transformation.

Proof

For genus 0 surface, removing one triangle is enough to apply the local version of our theorem, and to reconstruct the coordinates of all the vertices. Adding back the triangle introduces 3 dihedral angles that are completely fixed by the local integrability condition around any one of the 3 vertices. For higher genus models, the only difference with the local version is that some loops include combinations of the $2g$ nontrivial loops in addition to the one-ring loops, so we need to make the rotations match around them as well. In addition, unlike in the simply connected case, the translations around these non-local loops do not automatically match, hence the additional constraints. \square

Discussion As triangle inequalities do not eliminate DoFs, we can count the number of true DoFs as $2|E| - (3(|V| - 2) + 6 \times 2g)$ (corresponding respectively to the length and dihedral values, the local integrability conditions on all but two vertices, and the additional topological constraints). Using Euler's formula $|V| - |E| + |F| = 2(1 - g)$ and the fact that triangulated manifolds satisfy $3|F| = 2|E|$, we see that there are $3|V| - 6$ DoFs, matching exactly the number of DoFs of an immersed mesh up to rigid transformations. Note that the choice of the homology basis does not matter, since, roughly speaking, any basis will span the space of non-trivial loops.



Figure 1: From left to right: original and rearranged models of Octopus, bent Cactus, Armadillo, and Bunny, deformed using our linear reconstruction with position and optional rotation constraints. Octopus was deformed by relocating multiple handles (red) simultaneously, and the other models were results of one or more steps of deformation, with each step using one fixed handle (red) and one movable handle (yellow). Note that our method is robust even for the highly irregular triangle shapes in the Cactus model.

3. Immersion Energy through the Lift to 6D

Ideally, the integrability condition should be used as a constraint in the geometry processing procedure producing the discrete fundamental forms. However, in practice, we may have additional point and direction constraints imposed by the user, or conflicting fundamental form data due to various reasons; one such source of incompatibility could be the discretization itself. In this section, we propose a simple energy functional, the minimization of which leads to a globally optimal reconstruction in the sense that the local differential properties $F^{-1}dF$ deviates from the prescribed invariants in Ω as little as possible. This is equivalent to minimizing $dF - F\Omega$ provided that f is orthonormal. In the continuous setting, one could use

$$\int_M \|dF - F\Omega\|^2 = \int_M \|df - f\Omega_R\|^2 + \int_M \|d\mathbf{x} - f\Omega_x\|^2,$$

for such an energy, where $\|\cdot\|$ denotes the Frobenius norm of matrices. In the discrete setting, as we store different components of the lift at different locations, we can simply treat the rotation part and the translation part separately as above. To derive the discrete energy, we use the notation of discrete differential forms [DKT06]. We treat each entry of df and $f\Omega_R$ as dual 1-forms, and each entry in $d\mathbf{x}$ and $f\Omega_x$ as primal 1-forms. The discrete 1-forms are represented by their line integral along edges. Along a dual edge connecting face i and face j , the line integral of the exact form df is simply $f_j - f_i$, and the path-dependent line integral of the 1-form $f\Omega_R$ is approximated by $f_i(R_{ij} - Id)$. The primal 1-form $d\mathbf{x} - f\Omega_x$ can be integrated along the primal edge in each triangle. For any primal one-form β , the L^2 -norm is discretized to $\beta^T *_1 \beta$, where $*_1$ is the primal Hodge star, and for dual one-form γ , it is $\gamma^T (*_1)^{-1} \gamma$. Thus, we arrive at

the following weighted least-squares energies

$$E_f(f) = \frac{1}{2} \sum_{e \in E, e = T_i \cap T_j} w_e^{-1} \|f_j - f_i R_{ij}\|^2,$$

and

$$E_{\mathbf{x}}(x, f) = \frac{1}{2} \sum_{e=v_m v_n \in E} \sum_{T \ni e} w_{e,T} \|\mathbf{x}_n - \mathbf{x}_m - f_T \begin{pmatrix} a_{mn,T}^1 \\ a_{mn,T}^2 \\ 0 \end{pmatrix}\|^2,$$

where the w_e 's are the cotangent edge weights of the usual discrete Laplacian [PP93]. We can thus construct the mesh by minimizing the following energy (quadratic with respect to (x, f))

$$E_M(x, f) = w E_f(f) + E_{\mathbf{x}}(x, f), \quad (5)$$

where w is relative weighting (set to 1,000 in our tests). Varying w leads to a change in the “stiffness” of the mesh, with smaller w preserving the metric I better and larger w preserving dihedral angles better. However, the change in the result is subtle unless the weighting is changed in orders of magnitude.

The resulting linear system will be positive-semi-definite, and after fixing one vertex and one frame, the system becomes positive-definite. The cotangent weights used in E_f are thresholded to be above a small positive number (set to be 10^{-11} in our test) to improve the condition number. Without thresholding, the Cholesky factorization in the linear solver may fail. However, the threshold value does not lead to visually noticeable change on the results even if we increase it to 10^{-3} .

We can prove that the reconstruction does not depend on the choice of 2D local frame. Suppose we perform a random 2D rotation with angle θ_i within each face i . Given the same fundamental forms, the new transition rotation will become

$\hat{R}_{ij} = R_z(-\theta_i)R_{ij}R_z(\theta_j)$, and the new \hat{a} satisfies

$$\begin{pmatrix} \hat{a}_{ij,T}^1 \\ \hat{a}_{ij,T}^2 \\ 0 \end{pmatrix} = R_z(-\theta_i) \begin{pmatrix} a_{ij,T}^1 \\ a_{ij,T}^2 \\ 0 \end{pmatrix}.$$

Hence, $\|f_j - f_i \hat{R}_{ij}\|^2 = \|f_j R_z(-\theta_j) - f_i R_z(-\theta_i) R_{ij}\|^2$, since the Frobenius norm satisfies $\|AR\|^2 = \text{tr}(AR(AR)^T) = \text{tr}(AAT) = \|A\|^2$ for any orthogonal matrix R . Thus if the rotation part of the minimizer of the new energy is \tilde{f}_i , $\tilde{f}_i R_z(-\theta_i)$ is the minimizer of the original energy, and the translation part \mathbf{x} remains the same. Note that the above argument holds whether f is orthonormal or not.

4. Algorithm

The above convex energy function, together with possible constraints, can be optimized through a linear system resembling a Poisson equation. If we arrange elements of f in row-major order, we can express the sparse symmetric linear system using a block matrix; for example, the diagonal entry for the n -th row for \mathbf{x} would be $w_{nn}Id$, and the block on the i -th row and j -th column for f would be

$$-w_{ij} \begin{pmatrix} R_{ij}^T & 0 & 0 \\ 0 & R_{ij}^T & 0 \\ 0 & 0 & R_{ij}^T \end{pmatrix}.$$

We omit further discussions here, as all the coefficients can be deduced from the energy functional. Through its similarity to the Poisson equation, it is easy to see that this block matrix is positive-definite once we fix at least one position and one rotation (or other equivalent combinations of constraints) as mentioned above.

The main procedure for our algorithm is

1. Compute R_{ij} and a from I and II or directly from immersion when it is available.
2. Include hard or soft constraints for positions and (optionally) for rotations through variable elimination or penalty terms.
3. Solve the sparse linear system to get the immersion \mathbf{x} .

We do not enforce (and in fact cannot enforce with a linear system) that the frames stay orthonormal. However, as noted in [LSLCO05], since the R_{ij} 's are orthonormal, f remains close to being orthonormal, except for extreme constraints (in which case, one may (ortho)normalize the resulting f , and solve the system again with the normalized frames as soft constraints. In this optional step, the penalty weight on a frame for deviating from the orthonormal frame in the first solve is set to 1,000 in our tests. A larger penalty can lead to better preservation of the first fundamental form, but may also produce small discontinuity near constrained vertices). Due to the fact that the frames are stored on triangles, we can also easily derive constraints for f if needed, obtaining the face normal \mathbf{N} by the normalized cross product of two edges for triangles with all three vertices constrained.

5. Results

The basic local rigid-motion invariant formulation for surface equation can potentially benefit various applications involving geometry processing using triangle meshes. We developed a few proof-of-concept examples. We ran all the tests on an Intel Core2Dual@3.0GHz CPU with 4GB RAM.

Deformation Our first example is mesh deformation or editing using position and (optional) rotation constraints. Using Region of Interest and prefactorization of the SPD matrix, we can achieve interactive rates for moderate size regions, similar to e.g. [KCVS98, SCOL*04, LSLCO05]. See Figure. 1. We show some extreme test cases as in the survey paper [BS08] (Figure. 10) in Figure. 3.

Quasi-isometric Parameterization One interesting example when we manipulate the fundamental forms is to set II to 0, thus flattening the surface patch. In some sense, it is striving to preserve the first fundamental form, thus it can be called *quasi-isometric*. This method represents a new linear method to produce a parameterization with a “natural” boundary (see [DMA02]). In this case, orthogonality can be weakly enforced by a quadratic penalty term. We sometimes perform a second linear solve as for the extreme constraints, since the abrupt change in II leads to large deviations from the compatibility conditions. However, as such, it cannot guarantee an embedding in 2D, but just an immersion. In such cases, the target function still forces the frames to produce a planar mesh, but the frames may be turned left-handed, i.e., flipped from the original orientation for certain triangles. However, parameterization of typical meshes rarely gives rise to flips, as illustrated in Figure. 5.

Independence to Sampling Density In Figure. 2, a torus with varying vertex densities was deformed. As shown from the results, the overall shape remains the same despite the abrupt change in sampling density, and the symmetry of the shape is preserved.

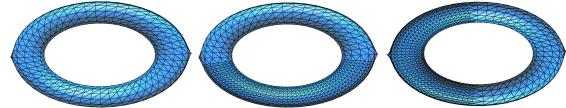


Figure 2: Torus with different sampling density produces similar deformation and preserves symmetry.

Comparison to Existing Work There are a number of differences between our method and the only existing linear and rotation-invariant approach [LSLCO05]. First, our discrete fundamental forms representation is (around 50%) more compact, independent of the choice of vertex normal, and independent of choice of the edges in the definition of frames (as we have proven mathematically, and observed numerically). Second, we do not require a prior embedding of the triangle mesh. Third, we couple the rotation and translation constraints in a natural fashion. As shown in Figure. 4, our method produces similar results measured by

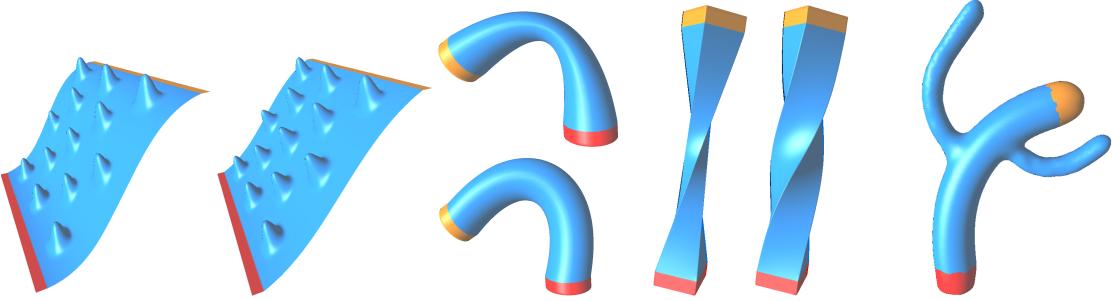


Figure 3: Test cases used in [BS08]. From left to right: bumpy plane before and after frame orthonormalization, 120° bent cylinder before (top) and after (bottom) orthonormalization, 135° twisted bar before and after orthonormalization, 70° bent cactus. Note that the results with orthonormalized frames are produced by a second linear solve using normalized columns of the frames in the first solve as soft constraints to handle these extreme cases. In these extreme deformations, the second solve produces visible differences except for the cactus case. In the bumpy plane test before normalization, the smooth transition between frames would force the bumps to point upward as the constraints force the frames at both handle regions to point upward.

edge length change, but produces less angle distortion, especially for the more complicated armadillo model. We added compatible rotation constraints for both methods, so that the linear rotation-invariant (LRI) coordinates [LSLC05] can produce results, although our method can in fact work with only translational constraints. For these large deformations, we found orthonormalization of the frames improves the results for both methods. We used a weight of 100 for the afore-mentioned soft constraints approach for orthonormalization of the frames in our method. Other surface deformation methods with point constraints are either not rotation-invariant or non-linear. For a comparison to these methods, refer to [BS08] (Figure. 10) for models deformed under same type of constraints (bumpy plane, cylinder bending, bar twisting, and cactus bending) as in Figure. 3.

6. Conclusion and Future Work

We have introduced compatibility conditions for edge lengths and dihedral angles for meshes with fixed topology, akin to Gauss's equation and the Codazzi equations. We have proposed a sparse linear system based on these lengths and angles, the solution of which produces a mesh immersed in 3D with edge lengths and dihedral angles close to the specified ones. We have also demonstrated the benefits of such a rotation-invariant representation for mesh deformation applications.

Our discretization mixes the face-based frame representation and intrinsic connection (ω_1^2) of [CDS10], the rotations induced by the second fundamental form (ω_1^3, ω_2^3) of [LSLC05], and the relation between deformation gradients and rotations (a^1, a^2) of [CPSS10].

One drawback of our approach (common to other work) is the lack of guarantee of frame orthonormality—although for certain applications, it can provide additional flexibility nec-

essary to satisfy the constraints. We wish to examine the possibility of using quaternions to represent rotations with a nonlinear but tractable optimization process as in, e.g., [CPS11].

Like the differential surface theory it is based on, our method cannot handle intersections without resorting to other methods. However, this is also common to other methods in the same category, and can actually be remedied by incorporating existing collision handling techniques.

We have proposed an ad-hoc method in Sec. 2.3 for conversions between matrix functions and discrete fundamental forms, and we wish to further investigate the continuous limits of the matrix functions when we refine the mesh. Another topic worth exploring is the design of a modified version of our energy functional to serve as a potential energy for thin shell simulations: we could associate translational velocities at vertices with a lumped mass matrix, and rotational velocities at faces with moments of inertia of each triangle. We also wish to study proper handling of potential energy with inhomogeneous and anisotropic material property using our basic representation. Based purely on geometric information, we can also try to weight different dihedral angles differently to enforce feature preservation, or weight regions differently to modify their rigidity according to user specifications. We can also manipulate the Frobenius norm, and put different relative weighting, e.g., for rotations and translations. The integrability conditions we proposed can potentially be combined in local neighborhoods to allow for local permissible editing operations on the rotation invariant representation that does not influence other areas—this seems plausible as a perturbation in a vertex location only changes the edge lengths and dihedral angles of its one-ring. We also wish to explore the application in deformation transfer using a patched-based extension to our representation similar to the extension to [LSLC05] in [BVGP09]. Last, we may

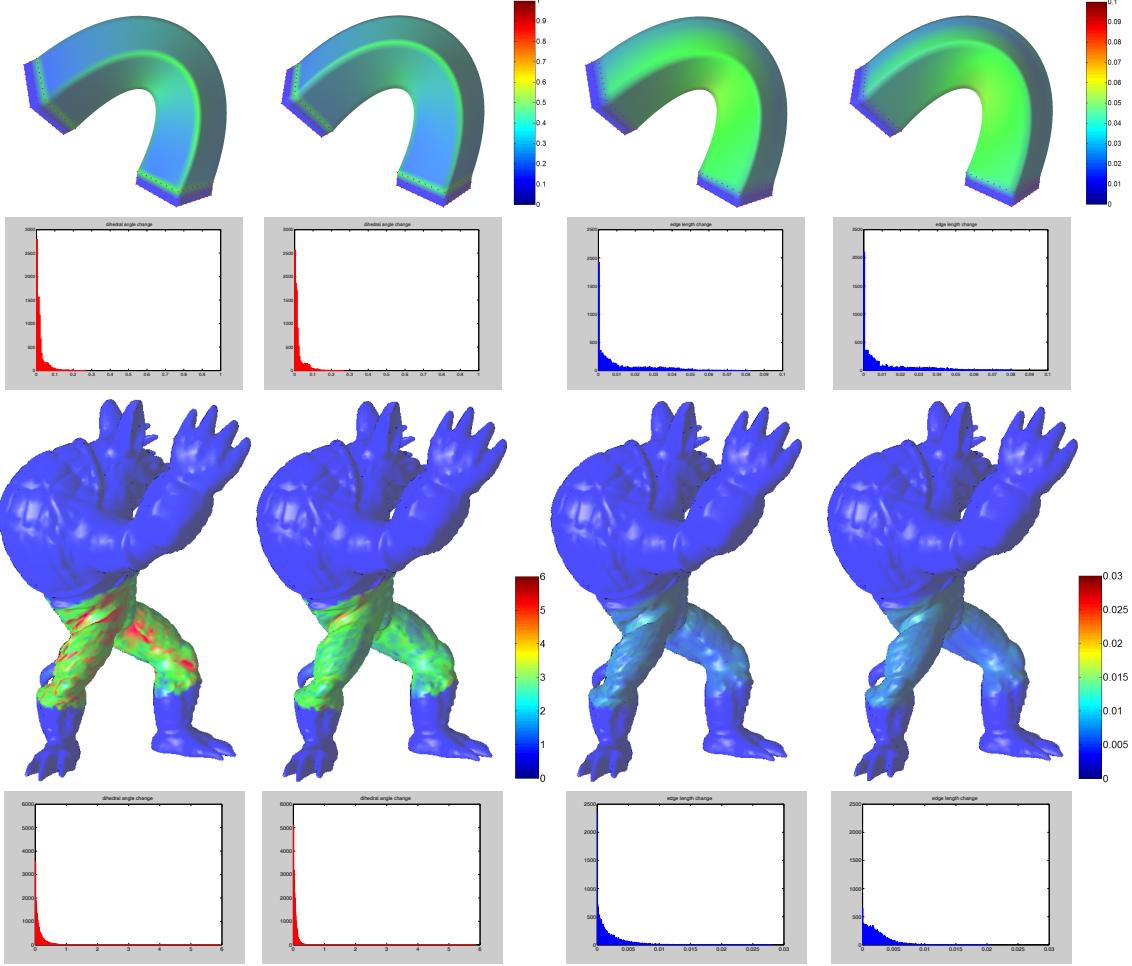


Figure 4: Visualization of distortion using pseudo-color and histograms. From left to right: angle distortion for results using LRI [LSLCO05], angle distortion for results using our method, edge distortion for results using LRI, and edge distortion for results using our method. The pseudo color denotes the absolute value of changes in edge length and dihedral angles measured in radian. The sizes of the bounding boxes for the bar model (with 1666 vertices) and the armadillo model (with 16K vertices) are $1 \times 1 \times 6$ and $1 \times 1.2 \times 0.9$, respectively.

try to enhance the efficiency of the algorithm through a multiscale approach, although it is unclear whether significant gains will be incurred since the large linear system we currently use is very sparse (each face is only adjacent to three other faces).

Acknowledgement

We wish to thank the reviewers and Mathieu Desbrun for their valuable suggestions. This work was supported in part by NSF grants IIS-0953096, CCF-0936830, CCF-0811313, and CMMI-0757123.

References

- [BK03] BENDELS G. H., KLEIN R.: Mesh forging: Editing of 3d-meshes using implicitly defined occluders. In *Symposium on*

Geometry Processing 2003 (June 2003), Eurographics Association. 2

[BK04] BOTSCHE M., KOBBELT L.: A remeshing approach to multiresolution modeling. In *Symposium on Geometry Processing 2004* (New York, NY, USA, 2004), SGP '04, ACM, pp. 185–192. 2

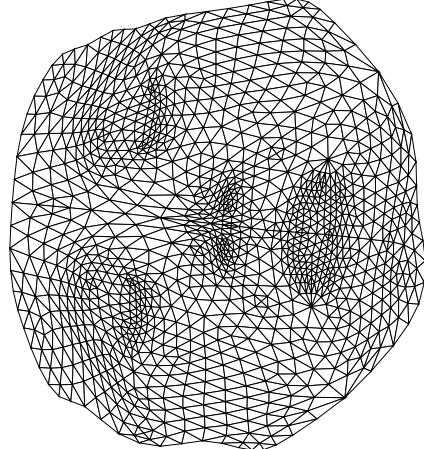
[BS08] BOTSCHE M., SORKINE O.: On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics* 14, 1 (2008), 213–230. 2, 7, 8

[BVG09] BARAN I., VLASIC D., GRINSPIUN E., POPOVIĆ J.: Semantic deformation transfer. *ACM Trans. Graph. (SIGGRAPH)* 28 (July 2009), 36:1–36:6. 8

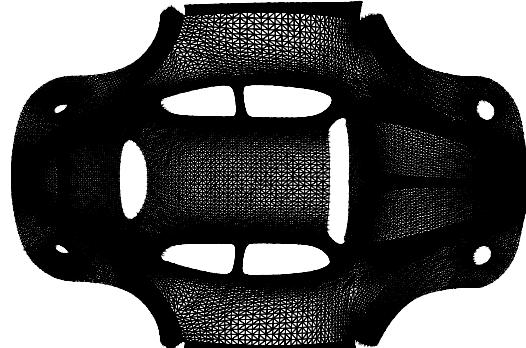
[CDS10] CRANE K., DESBRUN M., SCHRÖDER P.: Trivial connections on discrete surfaces. *Computer Graphics Forum (SGP)* 29, 5 (2010), 1525–1533. 4, 8

[CPS11] CRANE K., PINKALL U., SCHRÖDER P.: Spin transfor-

- mations of discrete surfaces. *ACM Trans. Graph. (SIGGRAPH)* 30 (August 2011), 104:1–104:10. 8
- [CPSS10] CHAO I., PINKALL U., SANAN P., SCHRÖDER P.: A simple geometric model for elastic deformations. *ACM Trans. Graph. (SIGGRAPH)* 29 (July 2010), 38:1–38:6. 2, 8
- [DKT06] DESBRUN M., KANSO E., TONG Y.: Discrete differential forms for computational modeling. In *ACM SIGGRAPH 2006 Courses* (New York, NY, USA, 2006), SIGGRAPH ’06, ACM, pp. 39–54. 6
- [DMA02] DESBRUN M., MEYER M., ALLIEZ P.: Intrinsic Parameterizations of Surface Meshes. *Computer Graphics Forum* 21 (2002). 7
- [FB11] FRÖHLICH S., BOTSCHE M.: Example-Driven Deformations Based on Discrete Shells. *Computer Graphics Forum* 30, 8 (Dec. 2011), 2246–2257. 2
- [Fra03] FRANKEL T.: *The Geometry of Physics: An Introduction, Second Edition*, 2 ed. Cambridge University Press, Nov. 2003. 2
- [IL03] IVEY T. A., LANDSBERG J. M.: *Cartan for Beginners: Differential Geometry Via Moving Frames and Exterior Differential Systems*, illustrated ed. American Mathematical Society, 2003. 2, 3
- [KCVS98] KOBBELT L., CAMPAGNA S., VORSATZ J., SEIDEL H.-P.: Interactive multi-resolution modeling on arbitrary meshes. In *Proceedings of SIGGRAPH 1998* (New York, NY, USA, 1998), ACM, pp. 105–114. 2, 7
- [KG08] KIRCHER S., GARLAND M.: Free-form motion processing. *ACM Trans. Graph.* 27 (May 2008), 12:1–12:13. 2
- [KVS99] KOBBELT L., VORSATZ J., SEIDEL H.-P.: Multiresolution hierarchies on unstructured triangle meshes. *Comput. Geom. Theory Appl.* 14, 1-3 (1999), 5–24. 2
- [LCOGL07] LIPMAN Y., COHEN-OR D., GAL R., LEVIN D.: Volume and shape preservation via moving frame manipulation. *ACM Trans. Graph.* 26 (January 2007). 2
- [LSCO*04] LIPMAN Y., SORKINE O., COHEN-OR D., LEVIN D., RÖSSL C., SEIDEL H.-P.: Differential coordinates for interactive mesh editing. In *Proceedings of Shape Modeling International* (2004), IEEE Computer Society Press, pp. 181–190. 2
- [LSLCO05] LIPMAN Y., SORKINE O., LEVIN D., COHEN-OR D.: Linear rotation-invariant coordinates for meshes. *ACM Trans. Graph.* 24 (July 2005), 479–487. 2, 4, 7, 8, 9
- [PKKG03] PAULY M., KEISER R., KOBBELT L. P., GROSS M.: Shape modeling with point-sampled geometry. *ACM Trans. Graph.* 22 (July 2003), 641–650. 2
- [PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics* 2 (1993), 15–36. 6
- [SCOL*04] SORKINE O., COHEN-OR D., LIPMAN Y., ALEXA M., RÖSSL C., SEIDEL H.-P.: Laplacian surface editing. In *Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing* (2004), Eurographics Association, pp. 179–188. 2, 7
- [WBH*07] WARDETZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *Comput. Aided Geom. Des.* 24 (November 2007), 499–518. 2
- [WDAH10] WINKLER T., DRIESEBERG J., ALEXA M., HORRMANN K.: Multi-scale geometry interpolation. *Computer Graphics Forum* 29, 2 (May 2010), 309–318. Proceedings of Eurographics. 2
- [YZX*04] YU Y., ZHOU K., XU D., SHI X., BAO H., GUO B., SHUM H.-Y.: Mesh editing with poisson-based gradient field manipulation. *ACM TRANS. GRAPH (SIGGRAPH)* 23 (2004), 644–651. 2
- [ZHS*05] ZHOU K., HUANG J., SNYDER J., LIU X., BAO H., GUO B., SHUM H.-Y.: Large mesh deformation using the volumetric graph laplacian. *ACM Trans. Graph. (SIGGRAPH)* 24 (July 2005), 496–503. 2
- [ZRKS05] ZAYER R., RÖSSL C., KARNI Z., SEIDEL H.-P.: Harmonic guidance for surface deformation. In *Eurographics* (2005), vol. 24:3 of *Computer Graphics Forum*, pp. 601–609. 2



(a) Nefertiti



(b) Beetle

Figure 5: Parameterization produced by setting the dihedral angles to zeros.