Graphical Models 76 (2014) 457–467

Contents lists available at ScienceDirect

Graphical Models

journal homepage: www.elsevier.com/locate/gmod

As-rigid-as-possible spherical parametrization

Chunxue Wang, Zheng Liu *, Ligang Liu

School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, China

ARTICLE INFO

Article history:
Received 1 March 2014
Received in revised form 25 March 2014
Accepted 28 March 2014
Available online 13 April 2014

Keywords:
Spherical parametrization
As-rigid-as-possible energy
Mesh processing

ABSTRACT

In this paper, we present an efficient approach for parameterizing a genus-zero triangular mesh onto the sphere with an optimal radius in an *as-rigid-as-possible* (ARAP) manner, which is an extension of planar ARAP parametrization approach to spherical domain. We analyze the smooth and discrete ARAP energy and formulate our spherical parametrization energy from the discrete ARAP energy. The solution is non-trivial as the energy involves a large system of non-linear equations with additional spherical constraints. To this end, we propose a two-step iterative algorithm. In the first step, we adopt a local/global iterative scheme to calculate the parametrization coordinates. In the second step, we optimize a best approximate sphere on which parametrization triangles can be embedded in a rigidity-preserving manner. Our algorithm is simple, robust, and efficient. Experimental results show that our approach provides almost isometric spherical parametrizations with lowest rigidity distortion over state-of-the-art approaches.

1. Introduction

Parametrization is essential for geometry processing and has wide applications in various fields, including texture mapping, texture synthesis, detail transfer, mesh completion, remeshing, surface approximation, scattered data fitting, and morphing, etc.

The problem of mesh parametrization is to compute a one-to-one mapping from a given mesh to a suitable domain. The most commonly used mesh is triangular mesh, and the mappings are required to be at least piecewise linear, so we only need to compute the vertex coordinates.

The parametrization of a closed genus-zero manifold triangular mesh preferably is done on its natural domain: the spherical domain. Many applications are quite sensitive to discontinuities in the parametrization, or cannot tolerate them at all. The big advantage of the spherical domain over the planar one is that it allows for seamless, continuous parametrization of genus-zero models, and there are a large number of such models in use [1].

As we have known, the common goal of parametrization is to find a mapping which minimizes some metric distortion of the original mesh. There are various types of parametrization methodologies, such as authalic (area-preserving) mapping, conformal (angle-preserving) mapping, isometric (length-preserving) mapping, and some combination of these [2]. In this paper, we aim to parameterize a genus-zero triangular mesh onto a sphere in an as-rigid-as-possible (ARAP) manner.

Specifically, we extend the ARAP planar parametrization method [2,3] to spherical domain. In the planar case, the ARAP parametrization [2] minimizes an "intrinsic" deformation energy function which can be expressed in terms of the singular values of the Jacobian of the parametrization. Unfortunately, the extension to the sphere is not straightforward. Furthermore, in *sharp* contrast to the planar case, we need to optimize a radius of the sphere on which parametrization triangles can be embedded on in a rigidity-preserving manner.

* Corresponding author.
E-mail address: liu.zheng.jojo@gmail.com (Z. Liu).

http://dx.doi.org/10.1016/j.gmod.2014.03.016
1524-0703/

After we formulate the smooth ARAP spherical parametrization energy, we propose the discrete variational description of the energy. We use a two-step iterative algorithm to solve the non-linear problem with additional spherical constraints. An optimal radius of the sphere is obtained to guarantee the rigidity of the parametrization. Experimental results show that our ARAP parametrization approach always gives the lowest rigidity distortion compared with state-of-the-art approaches [4–7].

2. Related work

2.1. Spherical parametrization

The problem of mesh spherical parametrization is mapping a piecewise linear surface with a discrete representation onto a spherical surface. There are many methods of spherical parametrization proposed in the past years. We refer the interested reader to [8,1] for a survey of the state-of-the-art in spherical parametrization research. Many of these methods are very similar to those of mapping simple meshes onto planar domain, whereas some of the linear methods become non-linear versions.

The simplest way to map a closed triangular mesh to the sphere is to reduce the problem to the planar case. Haker et al. [9] used a method which mapped the given genus-zero mesh into the plane and then used stereographic projection to map it to the sphere. Thus, it is more natural to parameterize the mesh directly on the sphere without going back and forth to the plane. Several methods for direct parametrization on the sphere exist. Gotsman et al. [5] showed a nice relationship between spectral graph theory and spherical parametrization, and embedded simple meshes onto the sphere by solving a quadratic system. Unfortunately, there was no analysis of the degrees of freedom in the various spherical embedding, the system was also quite slow even for a few hundred vertices. Saba et al. [10] provided an efficient optimization method combined with an algebraic multigrid technique to get the solution of the large system of non-linear equations in [5]. Li et al. [6] minimized the discrete harmonic energy to map a genus-zero surface to the unit sphere with good shapes which was suitable for surface fitting with PHT-splines.

Some spherical parametrization approaches have focused on directly optimizing the metric distortion. These approaches require extensive computation, due to the distortion measures are usually highly non-linear. Gu and Yau [11] gave an important point that a harmonic spherical map is conformal, then they proposed an iterative method which approximated a harmonic map without splitting. Other spherical conformal mappings are proposed in [4,12,13]. Sheffer et al. [14] proposed a method which measured angle distortion directly. They formulated a set of necessary and sufficient conditions for the spherical angles of the triangulation to form a valid spherical triangulation. The resultant mappings of all methods above are almost angle-preserving. However, the area distortion of these methods should be very large, and the area distortion is necessary to consider. Praun and Hoppe [15] used a coarse-to-fine solving scheme to iteratively optimize the $\mathcal{L}_2$ stretching energy [16] defined piecewise on the triangluar mesh. Zayer et al. [17] proposed a curvilinear spherical parametrization which better reduced area-distortion efficiently. Wan et al. [7] utilized the distortion energy [18] and presented an efficient hierarchical optimization scheme minimizing angle and area distortions. However, we find that little work has been done on rigidity-preserving in spherical parametrization, thus we propose a method using the ARAP energy to parameterize the surface in a rigidity-preserving manner.

Recently, some basic flows have been used to evolve surface geometry. Both are classic mean-curvature flow derivations (see [19] for a modern treatment of the continuous formulation, and [20] for the finite-elements discretization) [21] modified the flow and presented empirical evidence that does it define a stable surface evolution for genus-zero surfaces, but that the evolution converges to a conformal parametrization of the surface onto the sphere. However, discrete surface Ricci flow theory was developed by Chow and Luo [22] and a computational algorithm was introduced in [23]. With spherical geometry, Ricci energy is not strictly convex but converges to a local optimum [22,24,25] presented a framework for spherical parametrization using Euclidean Ricci flow, facilitating efficient and effective surface mapping.

2.2. ARAP energy

The ARAP energy is very important in geometry processing and has wide applications in editing [3], morphing [26,27], simulation [28], and planar parametrization [2,29]. The energy measures the sum of distance between deformation differentials and their corresponding rotation group [28]. Chao et al. [28] analyzed the relationship between the ARAP energy and standard elastic energy which commonly used in simulation. Local rigidity can be seen as the governing principle of various surface deformation models, and Sorkine et al. [3] used the ARAP energy to preserve the local-details of vertex's one-ring neighbors for surface modeling. The principle of ARAP deformation was successfully applied to shape interpolation [26,27] to determine intermediate shape path such that the deformation from source to target appears as rigid as possible. Liu et al. [2] used the ARAP energy for planar parametrization of disk-topology surfaces, they parameterize the mesh to the plane in a rigidity-preserving manner for each individual triangle. Myles and Zorin [29] described a method for finding cone locations, and extended the ARAP parametrization of disk-topology surfaces to general surfaces, which yielded seamless parametrization with low metric distortion.

3. Contribution

We extend the ARAP planar parametrization of disk-topology surfaces [2] to spherical parametrization of closed genus-zero mesh. This poses the problem to find an optimal local transformation for each individual mesh element, then stitch the transformed elements into a coherent

sphere. Compared with [2] in 2D, where the local transformations are applied to individual triangles, we apply the local transformations to tetrahedrons in 3D. Moreover, we optimize the radius of the sphere for isometric mapping. The reason we update the radius is that, our algorithm cannot guarantee all parametrization triangles are congruent to the original ones (there are some rigidity distortions and our method is an as rigid as possible one), thus we need to update the radius correspondingly to decrease the ARAP energy which determines the rigidity distortion directly.

In this context, our contributions are:

1. We develop the model of mapping the genus-zero triangular mesh to the sphere with an optimal radius in a rigidity-preserving manner. Our spherical parametrization model bases on minimizing the ARAP energy in 3D with additional spherical constraints.
2. We analyze the ARAP energy from the smooth variational description to its corresponding discrete 2D and 3D case.
3. It is non-trivial to solve such a non-linear model with non-linear constraints, we propose a two-step iterative algorithm to efficiently solve it.

4. ARAP energy

In this section, we give out an analysis of the ARAP energy. Firstly, we introduce the smooth, variational description ARAP energy. Then, we give out the discretization of the smooth energy in 2D and 3D case respectively.

Given a smooth map $f : M \to N, f(\mathbf{v}) = \mathbf{u}$ describes the mapping of a reference configuration $M \subset \mathbb{R}^n$ into a deformed configuration $N \subset \mathbb{R}^n$. Likewise, a differential vector $d\mathbf{v}$ is mapped by the deformation gradient df, then we get the energy

$$E(f) = \frac{1}{2}\int_M \min_{\mathbf{R}\in\mathbf{SO}(n)} |df - \mathbf{R}|^2. \tag{1}$$

The energy (1) measures the distance between the deformation differential df and the nearest rotation $\mathbf{R}$, characterizing how far f is from an isometry [28].

For convenience, we deNote 2-manifold simplicial complex as $M = (V, E, F)$ and 3-manifold simplicial complex as $M = (V, E, F, T)$.

4.1. The 2D case

For 2-manifold simplicial complex consisting of vertices, edges, triangles, the integral over M becomes a sum over triangles. Assume the triangles F of the mesh M are numbered with $t = 1$ to n (n is the number of triangles F). The original triangle t denotes $\mathbf{x}_t = \{\mathbf{x}_t^0, \mathbf{x}_t^1, \mathbf{x}_t^2\}$, and $\mathbf{u}_t = \{\mathbf{u}_t^0, \mathbf{u}_t^1, \mathbf{u}_t^2\}$ its image under df.

The corresponding discrete energy in 2D is the piecewise linear Dirichlet energy [2,30] as

$$E(f) = \frac{1}{4}\sum_{t=1}^{n}\sum_{i=0}^{2} \cot(\theta_t^i)\|(\mathbf{u}_t^i - \mathbf{u}_t^{i+1}) - \mathbf{R}_t(\mathbf{x}_t^i - \mathbf{x}_t^{i+1})\|^2, \tag{2}$$

where θ_t^i is the angle opposite the edge $(\mathbf{x}_t^i, \mathbf{x}_t^{i+1})$ in the triangle whose vertices are $\mathbf{x}_t$ and superscripts are all modulo 3.

4.2. The 3D case

For 3-manifold simplicial complex consisting of vertices, edges, triangles and tetrahedrons, the integral over M becomes a sum over tetrahedrons. Assume the tetrahedrons T of the mesh M are numbered with $\mathcal{T} = 1$ to n (n is the number of tetrahedrons T).

The original tetrahedron $\mathcal{T}$ denotes $\mathbf{x}_\mathcal{T} = \{\mathbf{x}_\mathcal{T}^0, \mathbf{x}_\mathcal{T}^1, \mathbf{x}_\mathcal{T}^2, \mathbf{x}_\mathcal{T}^3\}$, and $\mathbf{u}_\mathcal{T} = \{\mathbf{u}_\mathcal{T}^0, \mathbf{u}_\mathcal{T}^1, \mathbf{u}_\mathcal{T}^2, \mathbf{u}_\mathcal{T}^3\}$ its image under df.

Following the cotan formula for tetrahedron [28,31], the corresponding discrete energy in 3D is as follows

$$E(f) = \frac{1}{12}\sum_{\mathcal{T}=1}^{n}\sum_{\{i,j\}\in\{0,1,2,3\}} \cot(\theta_\mathcal{T}^{ij})\|\mathbf{x}_\mathcal{T}^k - \mathbf{x}_\mathcal{T}^l\|\|(\mathbf{u}_\mathcal{T}^i - \mathbf{u}_\mathcal{T}^j) - \mathbf{R}_\mathcal{T}(\mathbf{x}_\mathcal{T}^i - \mathbf{x}_\mathcal{T}^j)\|^2, \tag{3}$$

where $\{k, l\} = \{0, 1, 2, 3\}\backslash\{i, j\}$, $\theta_\mathcal{T}^{ij}$ is the dihedral angle opposite edge $(\mathbf{x}_\mathcal{T}^i, \mathbf{x}_\mathcal{T}^j)$ in the tetrahedron (see Fig. 1).

5. Model

If each triangle of the 3D triangular mesh is required to be flattened onto the sphere independently of the other triangles, this would certainly be easy (see Fig. 2 (middle)), we denote the flattened triangles on the sphere with local isometric parametrization as $\mathbf{x}_t = \{\mathbf{x}_t^0, \mathbf{x}_t^1, \mathbf{x}_t^2\}$. Note that $\mathbf{x}_t$ depends on radius r of the sphere. However, requiring all the flattened triangles on the sphere fit together into one coherent mesh with correct orientations is a main challenge (see Fig. 2 (right)). Obviously some of the triangles are going to be deformed in this process, in this paper we only allow each triangle to be deformed by rigid transformation with additional spherical constraints. We aim to find a single parametrization of the entire mesh, i.e., a piecewise linear mapping from the 3D mesh to the 3D sphere, described by assigning 3D coordinates $\mathbf{u}$ to each of the n vertices on the sphere. For each triangle t, we denote the parametrization coordinates as $\mathbf{u}_t = \{\mathbf{u}_t^0, \mathbf{u}_t^1, \mathbf{u}_t^2\}$.

In 2D case, the planar parametrization of the ARAP energy was studied by [2] and its discrete setting is Eq. (2). Firstly they flattened each 3D triangle into the plane with

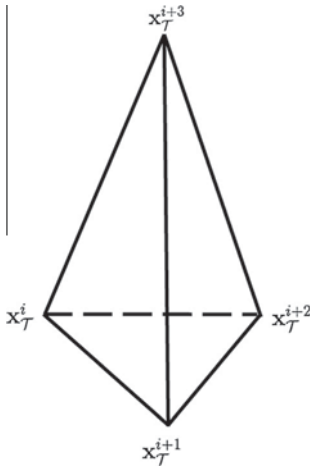

Fig. 1. The discrete ARAP energy in 3D corresponds to tetrahedrons.

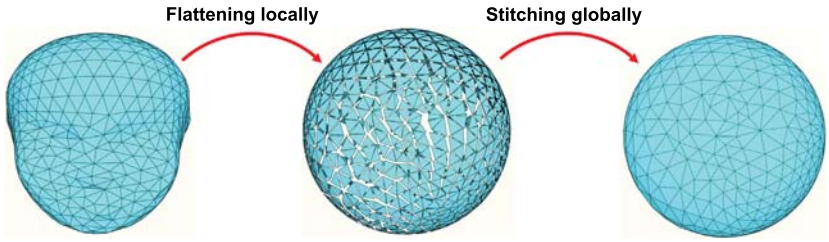

Fig. 2. Parameterizing a mesh by aligning local flattened triangles on the sphere in a rigidity-preserving manner. (Left) Original mesh; (middle) flattened triangles of original mesh on the sphere; (right) 3D ARAP spherical parametrization result.

its own local isometric parametrization, then used a local/global scheme to stitch the separated triangles into a coherent mesh.

In 3D case, however, with each flattened triangle $\mathbf{x}_t$ of the original mesh and its piecewise linear mapping result $\mathbf{u}_t$ of the parametrization of the entire mesh, we cannot calculate a unique 3×3 Jacobian matrix (deformation gradient *df*). In order to solve this problem, we add an auxiliary vertex $\mathbf{O}=\{0,0,0\}$ to each triangle of the mesh, thus the original mesh is converted to a 3-manifold simplicial complex. Then $\mathbf{x}_t$ corresponds to a tetrahedron $\mathbf{x}_{\mathcal{T}}=\{\mathbf{x}_t,\mathbf{O}\}=\{\mathbf{x}_t^0,\mathbf{x}_t^1,\mathbf{x}_t^2,\mathbf{O}\}$ and $\mathbf{u}_t$ corresponds to a tetrahedron $\mathbf{u}_{\mathcal{T}}=\{\mathbf{u}_t,\mathbf{O}\}=\{\mathbf{u}_t^0,\mathbf{u}_t^1,\mathbf{u}_t^2,\mathbf{O}\}$ (see Fig. 4). With the help of the auxiliary vertex, we can easily calculate the mapping f between $\mathbf{x}_{\mathcal{T}}$ and $\mathbf{u}_{\mathcal{T}}$, and use a 3×3 Jacobian matrix (deformation gradient *df*) to describe the mapping which is a constant per tetrahedron.

We define the ARAP spherical energy of the parametrization coordinates $\mathbf{u}_{\mathcal{T}}$ and the rotation group $\mathbf{R}_{\mathcal{T}}$ for each tetrahedron, consequently our problem is minimizing the energy as follows

$$\begin{aligned}&\min\quad E(\mathbf{u}_{\mathcal{T}},\mathbf{R}_{\mathcal{T}},r)=\sum_{\mathcal{T}}V_{\mathcal{T}}\|J(\mathbf{u}_{\mathcal{T}})-\mathbf{R}_{\mathcal{T}}\|_F^2,\\&\text{s.t.}\quad \|\mathbf{u}_{\mathcal{T}}^i\|^2=r^2,\quad i=0,1,2,\end{aligned}\tag{4}$$

where $V_{\mathcal{T}}$ is the volume of tetrahedron $\mathcal{T}$, $J(\mathbf{u}_{\mathcal{T}})$ is the 3×3 Jacobian matrix which depends on $\mathbf{u}_{\mathcal{T}}$ and r is the radius of the sphere.

Note that each tetrahedron $\mathbf{u}_{\mathcal{T}}$ consists of the triangle $\mathbf{u}_t$ with an additional constant vertex $\mathbf{O}$, since the rotation of $\mathbf{u}_{\mathcal{T}}$ is equal to the rotation of $\mathbf{u}_t$ ($\mathbf{R}_{\mathcal{T}}=\mathbf{R}_t$). We have mentioned above that $\mathbf{u}$ denote the 3D spherical parametrization coordinates of the vertices and $\mathbf{R}$ denote the rotation group of the triangles. Then, following [28,31] we can use the discrete 3D ARAP energy (3) to rewrite (4) (instead of in terms of the Jacobians) in an explicit form (in terms of the parametrization coordinates $\mathbf{u}$ and the mesh vertex coordinates $\mathbf{x}$)

$$\begin{aligned}E(\mathbf{u},\mathbf{R},r)=&\frac{1}{12}\sum_{t=1}^{n}\sum_{i=0}^{2}[\cot(\alpha_{\mathcal{T}}^{i,i+1})\|\mathbf{x}_t^{i+2}-\mathbf{O}\|\|(\mathbf{u}_t^i-\mathbf{u}_t^{i+1})\\&-\mathbf{R}_t(\mathbf{x}_t^i-\mathbf{x}_t^{i+1})\|^2+\cot(\beta_{\mathcal{T}}^{i,o})\|\mathbf{x}_t^{i+1}\\&-\mathbf{x}_t^{i+2}\|\|(\mathbf{u}_t^i-\mathbf{O})-\mathbf{R}_t(\mathbf{x}_t^i-\mathbf{O})\|^2]\\=&\frac{1}{12}\sum_{t=1}^{n}\sum_{i=0}^{2}[\cot(\alpha_{\mathcal{T}}^{i,i+1})\|\mathbf{x}_t^{i+2}\|\|(\mathbf{u}_t^i-\mathbf{u}_t^{i+1})\\&-\mathbf{R}_t(\mathbf{x}_t^i-\mathbf{x}_t^{i+1})\|^2+\cot(\beta_{\mathcal{T}}^{i,o})\|\mathbf{x}_t^{i+1}\\&-\mathbf{x}_t^{i+2}\|\|\mathbf{u}_t^i-\mathbf{R}_t\mathbf{x}_t^i\|^2]\end{aligned}\tag{5}$$

where $\alpha_{\mathcal{T}}^{i,i+1}$ is the dihedral angle opposite the edge $(\mathbf{x}_t^i,\mathbf{x}_t^{i+1})$ and $\beta_{\mathcal{T}}^{i,o}$ is the dihedral angle opposite the edge $(\mathbf{x}_t^i,\mathbf{O})$ in the tetrahedra. Note that we optimize a radius r of the sphere in the energy (5), for reason that the radius r directly influences the ARAP spherical parametrization result (see Section 6.2). It is non-trivial to solve such a non-linear problem with non-linear constraints, hence in next section we will provide a simple and efficient algorithm to deal with the problem.

6. Algorithm for ARAP spherical parametrization

In order to solve the minimization problem (4) for an ARAP spherical mapping, we present a two-step iterative algorithm by separating variables of the parametrization coordinates $\mathbf{u}$, the rigid rotation group $\mathbf{R}$ and the radius r.

Starting from an initial spherical parametrization from any available algorithm, our algorithm alternatively updates the parametrization coordinates $\mathbf{u}$, rotation group $\mathbf{R}$ and optimal radius r in two steps (see Fig. 3). The two-step iterative algorithm is described as follows.

6.1. Step one: the local/global algorithm

In step one, we fix the radius r of the sphere, and use a local/global iterative scheme to update the parametrization coordinates $\mathbf{u}$ and the rotation group $\mathbf{R}$. In the local phase of step one, the optimal rotation $\mathbf{R}_t(\mathbf{R}_{\mathcal{T}}=\mathbf{R}_t)$ is computed for per tetrahedron, assuming $\mathbf{u}$ are fixed. Then in the global phase, the rotation group $\mathbf{R}$ are assumed fixed, the optimal $\mathbf{u}$ are solved for as a sparse linear system.

6.1.1. Local phase

Remember that, in order to calculate the 3×3 Jacobian matrix (deformation gradient *df*), we add an auxiliary vertex $\mathbf{O}=\{0,0,0\}$ to each flattened triangle $\mathbf{x}_t$ and construct the corresponding tetrahedron $\mathbf{x}_{\mathcal{T}}$ (see Fig. 4). Since the optimal rotation of each flattened triangle is equal to the rotation of its corresponding tetrahedron, we minimize the following energy function for a single tetrahedron which can easily be deduced from energy (3)

$$E_{\mathbf{x}_{\mathcal{T}}}=\sum_{\{i,j\}\in\{0,1,2,3\}}\cot(\theta_{\mathcal{T}}^{ij})\|\mathbf{x}_{\mathcal{T}}^k-\mathbf{x}_{\mathcal{T}}^l\|\|(\mathbf{u}_{\mathcal{T}}^i-\mathbf{u}_{\mathcal{T}}^j)-\mathbf{R}_{\mathcal{T}}(\mathbf{x}_{\mathcal{T}}^i-\mathbf{x}_{\mathcal{T}}^j)\|^2,\tag{6}$$

where $\{k,l\}=\{0,1,2,3\}\backslash\{i,j\}$, $\theta_{\mathcal{T}}^{ij}$ is the dihedral angle opposite edge $(\mathbf{x}_{\mathcal{T}}^i,\mathbf{x}_{\mathcal{T}}^j)$ in the tetrahedron.

To calculate this, we rewrite the energy (6) as

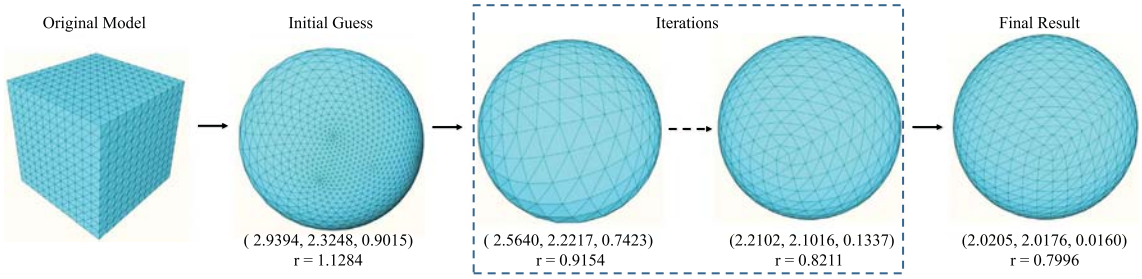

Fig. 3. Overview of our iterative algorithm. Spherical parameterizing a cube model. Starting from an initial guess from available algorithm, our algorithm runs a two-step processing to update the mesh. The final result is explicitly preserving the parametrization triangles as rigid as possible shown in the right. Numbers in bracket denote area, angle and rigidity distortion (the metric distortion defined in Section 8). The r denotes the optimal radius (see Section 6.2 for the optimal radius searching scheme) of the final parametrization result.

$$E_{\mathbf{x}_\mathcal{T}} = \sum_{i=0}^{5} w_i \|\mathbf{e}_i' - \mathbf{R}_\mathcal{T}\mathbf{e}_i\|^2, \tag{7}$$

where $w_i = \cot(\theta_\mathcal{T}^{i,j})\|\mathbf{x}_\mathcal{T}^k - \mathbf{x}_\mathcal{T}^l\|$ are the cotangent weights for tetrahedron, $e_i' = \mathbf{u}_\mathcal{T}^i - \mathbf{u}_\mathcal{T}^j$ and $e_i = \mathbf{x}_\mathcal{T}^i - \mathbf{x}_\mathcal{T}^j$.

Inspired by [2,3], this is solved as $\mathbf{R}_\mathcal{T} = \mathbf{V}_{\mathbf{x}_\mathcal{T}}\mathbf{U}_{\mathbf{x}_\mathcal{T}}^T$ by the signed SVD factorization of the cross-covariance matrix.

$$\mathbf{S}_{\mathbf{x}_\mathcal{T}} = \sum_{i=0}^{5} w_i \mathbf{e}_i \mathbf{e}_i'^T = \mathbf{U}_{\mathbf{x}_\mathcal{T}}\mathbf{D}_{\mathbf{x}_\mathcal{T}}\mathbf{V}_{\mathbf{x}_\mathcal{T}}^T. \tag{8}$$

6.1.2. Global phase

After we apply the rotation $\mathbf{R}_\mathcal{T}$ to each tetrahedron $\mathbf{x}_\mathcal{T}$ individually, the corresponding intermediate result may not coincide as shown in Fig. 2 (middle). Thus, we need to stitch them in the global phase.

Note that we fix radius r and rotation group $\mathbf{R}$ to solve parametrization coordinates $\mathbf{u}$ in the global phase, since the energy $E(\mathbf{u},\mathbf{R},r)$ (5) is quadratic in $\mathbf{u}$. The minimum $\mathbf{u}$ can be found by setting the gradient of (5) to zero and solving the associated linear sparse system.

To calculate this, overloading the notation slightly, we rewrite the energy function (5) in terms of the tetrahedrons:

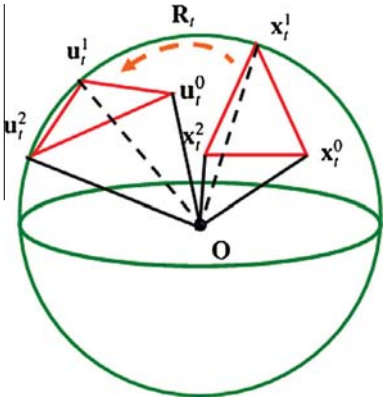

Fig. 4. In the local phase, we add an auxiliary vertex $\mathbf{O}$ to each triangle $\mathbf{x}_t$ to construct the corresponding tetrahedron $\mathbf{x}_\mathcal{T}$, and the optimal rotation $\mathbf{R}_\mathcal{T}$ is computed per tetrahedron assuming the $\mathbf{u}$ are fixed.

$$E(\mathbf{u},\mathbf{R},r) = \frac{1}{12}\left[\sum_{(i,j)\in he} w_{ij}\|(\mathbf{u}_i - \mathbf{u}_j) - \mathbf{R}_{\mathcal{T}_{(i,j)}}(\mathbf{x}_i - \mathbf{x}_j)\|^2 + \sum_i\sum_k w_{io}\|\mathbf{u}_i - \mathbf{R}_{\mathcal{T}_{ve(k)}}\mathbf{x}_i\|^2\right], \tag{9}$$

where he is the set of half-edges in the original mesh; $\mathbf{u}_i$ and $\mathbf{x}_i$ are coordinates of vertices i; $\mathcal{T}_{(i,j)}$ is the tetrahedron containing the triangle with half-edge (i,j); w_{ij} and w_{io} are the tetrahedron cotan weights which are the dihedral angles opposite edges $((i,j),(i,o))$ time edge lengths opposite edges $((i,j),(i,o))$; ve is the set of edges of tetrahedrons which contain the edge $(i,\mathbf{O})$; $\mathcal{T}_{ve}$ is the set of tetrahedrons containing the edge $(i,\mathbf{O})$ and $\mathbf{R}_{\mathcal{T}_{ve}}$ is the rotation group corresponding to the tetrahedrons $\mathcal{T}_{ve}$. N_{ve} is the number of edges in ve and $k \in N_{ve}$.

Setting the gradient of (9) to zero, we obtain the following set of linear equations for $\mathbf{u}$

$$\sum_{j\in\mathcal{N}(i)}(w_{ij}+w_{ji})(\mathbf{u}_i - \mathbf{u}_j) + \sum_{(i,\mathbf{O})\in ve} w_{io}\mathbf{u}_i = \sum_{j\in\mathcal{N}(i)}(w_{ij}\mathbf{R}_{\mathcal{T}(i,j)} + w_{ji}\mathbf{R}_{\mathcal{T}(j,i)})(\mathbf{x}_i - \mathbf{x}_j) + \sum_{(i,\mathbf{O})\in ve} w_{io}\mathbf{R}_{\mathcal{T}_{ve(k)}}\mathbf{x}_i. \tag{10}$$

The entries of the associated matrix depend only on the geometry of the input 3D mesh. Thus this sparse matrix is fixed throughout the step one (the local/gobal step), allowing us to pre-factor it (e.g. with Cholesky decomposition) and reuse the factorization many times in order to accelerate the process. This has a significant impact on algorithm efficiency.

Once we compute $\mathbf{u}$ from (10), all the triangles almost distribute nearly the surface of sphere. Heuristically, we project them to the sphere of radius r along radius direction. That is, we get the new spherical parametrization satisfying the model (4).

6.2. Step two: the optimal r searching

In this section, we will talk about how to find the optimal radius r. The chosen of the r is important, since it determines the rigidity distortion directly. If r is too small or too large, it will increase the rigidity distortion sharply. Because of aiming to parameterize the triangles in a rigid-

ity preserving manner, we try to keep the total area of the mesh as the original one. As a consequence, we set the initial r_0 just as below

$$r_0 = \sqrt{\frac{\sum_t A_t}{4\pi}},$$

where A_t is the area of the triangle t. Unfortunately, our algorithm cannot guarantee that all parametrization triangles are congruent to their original ones. The total area of the parametrization mesh is changed during the iterative process, thus we need to update the radius during the iterations correspondingly. We update the radius as follows

$$r_{k+1} = \sqrt{\frac{S_{k+1}}{S_k}} r_k, \tag{11}$$

where r_k is the *k-th* iteration of r, S_k is the total area of *local/global* results in *k-th* iteration instead of projected results in *k-th* iteration.

6.3. Algorithm

Algorithm 1. ARAP spherical parametrization

Input: The triangular mesh $M = \{E, F, V\}$
Output: The spherical parametrization coordinates **u**
Initialization: Generate an initial parametrization coordinates $\mathbf{u}_0$ and a radius r_0.
Repeat until convergence {
Step one:
Repeat until convergence {
Local phase: Update rotation group **R** with Eq. (8) (Section 6.1.1)
Global phase: Update parametrization **u** with Eq. (10) (Section 6.1.2) }
Step two:
Update optimal radius r with Eq. (11) (Section 6.2)
}

Advantages over previous methods. Our two-step iterative approach can preserve the parametrization triangles as rigid as possible and get an optimal approximate sphere for isometric mapping. Compared with previous spherical parametrization methods [5,11,6,7], our method can yield the best rigidity-preserving results. In contrast to the conformal method [11], our method has a very small, even insignificant penalty in angle distortion. Compared with the hierarchical method [7], our method has a small penalty in area distortion. However, if we consider the value of area and angle distortion synthetically, our ARAP parametrization consistently gives the best results (See Section 8).

7. Discussion

We discuss about some issues of our algorithm in this section including initialization, tessellation, convergence, flipping and complex model processing.

7.1. Initialization and tessellation

Our algorithm requires an initial parametrization result to start it off. The initial parametrization should be a valid embedding (contains no flips) with an initial radius r_0. We have applied the initial parametrization produced by some available algorithms, such as the convex parametrization [5], the conformal parametrization [11] and the harmonic parametrization [6]. We tested the sensitivity of our algorithm to different types of initializations, and found that the parametrization results of our algorithm are insensitive to the choosing of the initializations. Fig. 5 shows the same parametrization results of our algorithm by inputting different initializations. However, the time of convergence is impacted by the initialization. The convergence is slow for a significantly distorted initialization, while fast for a good initialization.

To evaluate that whether our algorithm is sensitive to tessellation, we have applied a model with different quality. Fig. 6 shows almost the same distortions are obtained with different mesh quality.

7.2. Convergence

The 3D ARAP energy (5) has been minimized by an alternating solver referred in Section 6. Although it is difficult to formally prove convergence of the two-step iterative algorithm, the energy in each step is guaranteed to reduce, thus this energy will eventually converge. And we prove the convergence of our algorithm via numerical experiments. We show ARAP energy curves of the models in Fig. 10 for an example. Fig. 7 shows that the energy always decreases in each iteration and finally converges to a stable minimum.

7.3. Flipping and complex model processing

In a valid parametrization result, the flipped triangles should be eliminated. Our algorithm uses a global Poisson equation to stitch the local elements into a coherent one, then projects the stitched result onto sphere.

Unfortunately, the projection operation can not guarantee the parametrization result without flips, especially for highly curvature sections, such as the long and thin protrusion regions. In this case, we sacrifice some rigidity of the extreme regions to obtain a valid spherical embedding. We solve this problem with a final post-processing phase using the harmonic energy of [6] to deal with the extreme regions. That is, we fix all non-flipped triangles and recompute the parametrization coordinates of the flipped triangles using the scheme in [6]. We have applied our method on the large and complex model, the hand model with many highly curvature sections, to verify the efficiency of it (see Fig. 8).

Since in most cases, there are only a few flips, sprinkled throughout the parametrization, the post-processing removes the flips without changing much else. Note that *all* the spherical parametrization results in this paper are without *any* post-processing and contain no flipped triangles, except the ones in Fig. 8 to demonstrate the flipping and complex model processing.

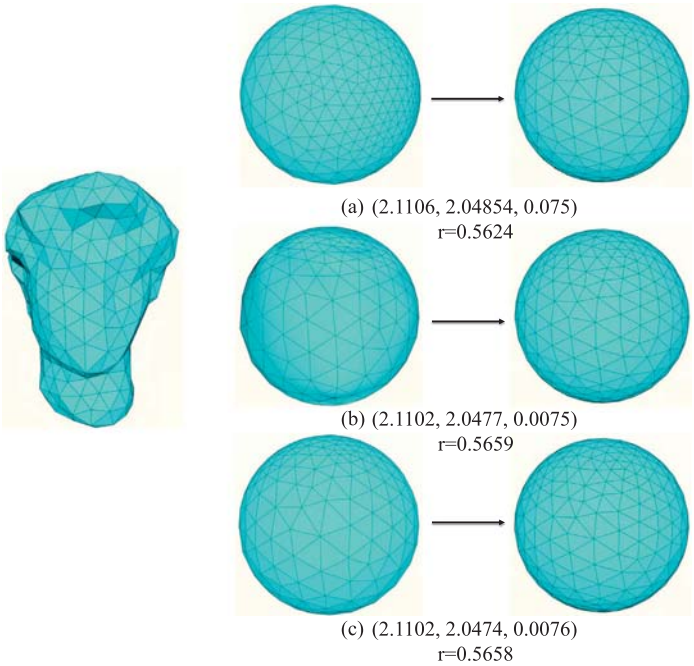

Fig. 5. Comparison of our parametrization results with different initial guesses. (a) Initialized with the convex parametrization [5]; (b) initialized with the harmonic parametrization [6]; (c) initialized with the conformal parametrization [11]. Numbers in bracket denote area, angle and rigidity distortion. The r denotes the optimal radius.

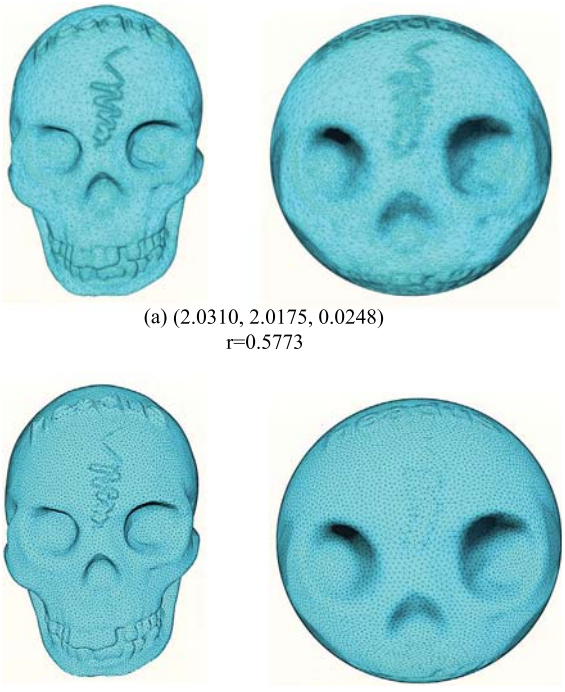

Fig. 6. The same model with different mesh quality produce approximate metric distortions. (a) The original Skull model and its ARAP parametrization result; (b) the re-meshed Skull model and its ARAP parametrization result. Numbers in bracket denote area, angle and rigidity distortion. The r denotes the optimal radius.

8. Experimental results and comparison

8.1. Distortion

To quantify the parametrization distortion, we compute area, angle and rigidity metric distortion following [2]. We flatten each triangle t of original mesh and parametrization mesh into the same plane, then compute the signed singular values $\sigma_{1,t}$ and $\sigma_{2,t}$ of the Jacobians J_t for triangle t, as defined in [32,33,2]:

$$D^{area} = \sum_t \rho_t(\sigma_{1,t}\sigma_{2,t} + 1/(\sigma_{1,t}\sigma_{2,t})),$$

$$D^{angle} = \sum_t \rho_t(\sigma_{1,t}/\sigma_{2,t} + \sigma_{2,t}/\sigma_{1,t}),$$

$$D^{rigidity} = \sum_t \rho_t((\sigma_{1,t} - 1)^2 + (\sigma_{2,t} - 1)^2),$$

where the weight ρ_t is

$$\rho_t = A_t \Big/ \sum_t A_t,$$

where A_t is area of triangle t.

8.2. Implementation details

We have implemented our two-step iterative spherical parametrization algorithm using MATLAB on a Pentium 4 2.16 GHz laptop with 2 GB RAM. In step one, we use an iterative local/global method to calculate the parametrization coordinates **u**. In the local phase we used the standard SVD implementation (used for polar decomposition of 3×3

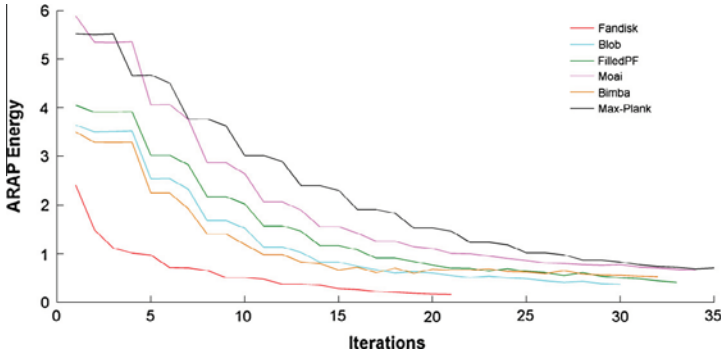

Fig. 7. The ARAP energy curves of the models in Fig. 10. These curves show that the rigidity energy always decreases in each iteration and finally converges to a stable minimum.

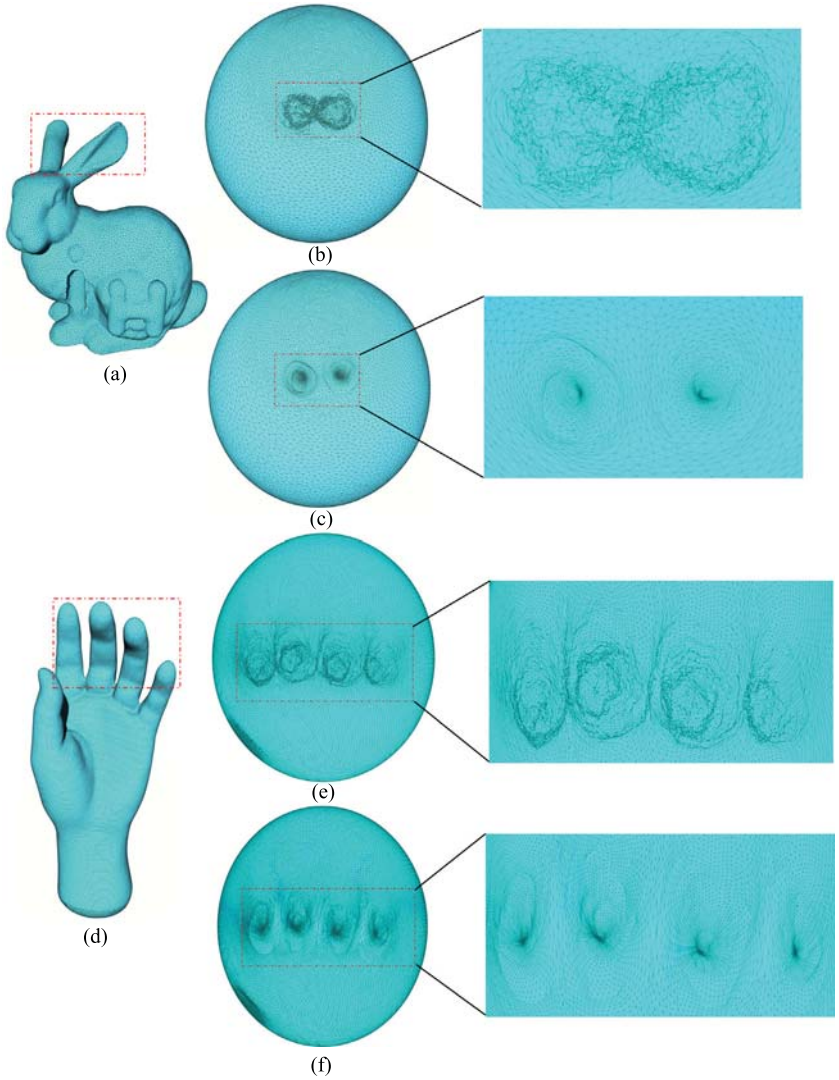

Fig. 8. Flipping and complex model processing. (a) The original Bunny model. (b) The parametrization result of our ARAP approach. Note that there are some flips corresponding to the region of Bunny's ears. (c) The post-processing result of the Bunny model without any flips using harmonic energy [6]. (d) The complex Hand model with many highly curvature sections. (e) The parametrization result of our ARAP approach. Note that there are some flips corresponding to the fingers of the hand. (f) The post-processing result of the Hand model without any flips using harmonic energy [6].

Table 1
Model statistic and performance, v denotes the number of vertices, f denotes the number of faces, i denotes the initialization to start our algorithm off, radius denotes the optimal radius and s denotes the convergence time in seconds. We use different initializations from the convex method [5], the harmonic method [6] and the conformal method [11] to start our algorithm off, and record the time of convergence. Note that the time of convergence is impacted by different initializations.

Model	v/f	Radius			Running time (s)		
		Convex	Harmonic	Conformal	Convex	Harmonic	Conformal
Head	647/1290	0.5781	0.5867	0.5865	102.33	134.32	118.35
Cube	866/1728	0.7996	0.7995	0.7936	182.71	164.61	156.62
David	361/718	0.5624	0.5659	0.5658	12.66	10.19	10.21
Max-Planck	15000/29996	/	0.6543	0.6532	/	798.67	655.93
Fandisk	606/1208	0.3387	0.3237	0.3347	13.68	13.17	10.79
Blob	2000/3996	/	0.4500	0.4483	/	69.87	62.61
FilledPF	2014/4028	/	0.4724	0.4744	/	120.91	114.61
Moai	2000/3996	/	0.5042	0.5035	/	53.45	49.92
Bimba	30002/59999	/	0.3848	0.3831	/	726.90	654.87
Bunny	20002/40000	/	0.5329	/	/	708.89	/
Hand	40268/84132	/	0.3541	/	/	1246.84	/

matrices). The most time-consuming part of our algorithm is solving the sparse linear system (5) in the global phase. The factorization of the normal equation may take a longer time. At each iteration step, only back-substitutions are performed to solve the system. We used the sparse Cholesky linear solver for matrix computation. Table 1 gives the model statistic and the time required of the parametrization results. Note that the convex method [5] involves a solution of a large system of non-linear equations. For large geometry models, we use conformal or harmonic parametrization result as initialization.

8.3. Parametrization results

We have applied our approach to parameterize a variety of 3D genus-zero meshes and compared with other relevant methods including the conformal parametrization [11], the harmonic parametrization [6], the convex parametrization [5] and the hierarchical parametrization [7]. Note that we normalize the bounding box of each model to unit length for comparison's sake. We show some results in Figs. 9, 10.

Fig. 9 compares the parametrization results of a Max-Planck model with various approaches. And Fig. 10 shows the parametrization results of different models.

The values of the area, angle and rigidity distortion measures obtained by various algorithms are summarized in Tables 2,3, respectively.

From the metric distortion we can see that

- As is evident our ARAP parametrization consistently gives the best values of the rigidity distortion.

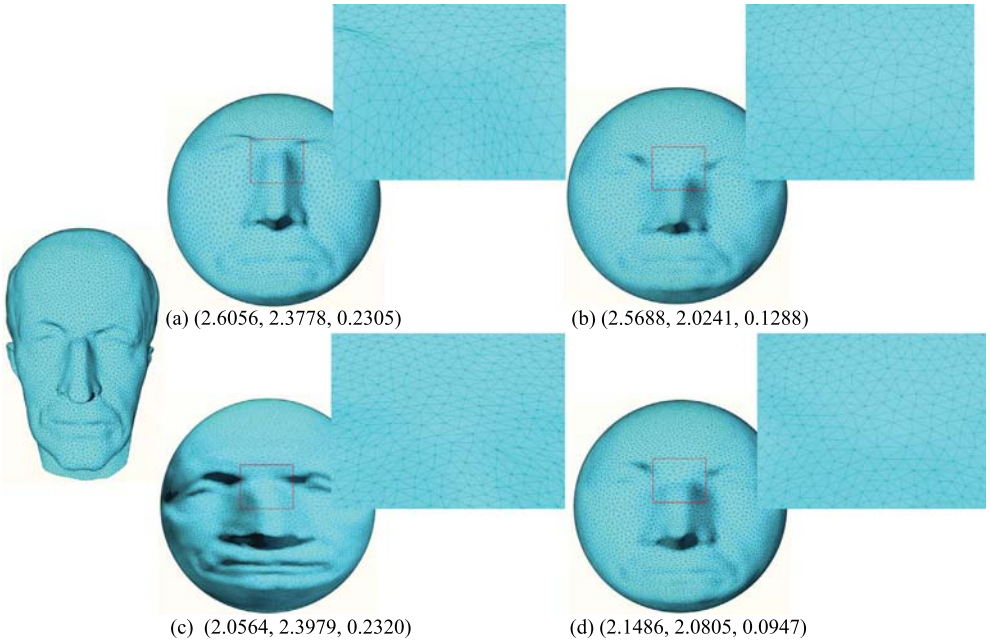

Fig. 9. Results of parameterizing the Max-Planck model (leftmost) using different approaches. (a) The harmonic parametrization result [6]; (b) the conformal parametrization result [11]; (c) the hierarchical parametrization result [7]; (d) our result. Numbers in bracket denote area, angle and rigidity distortion.

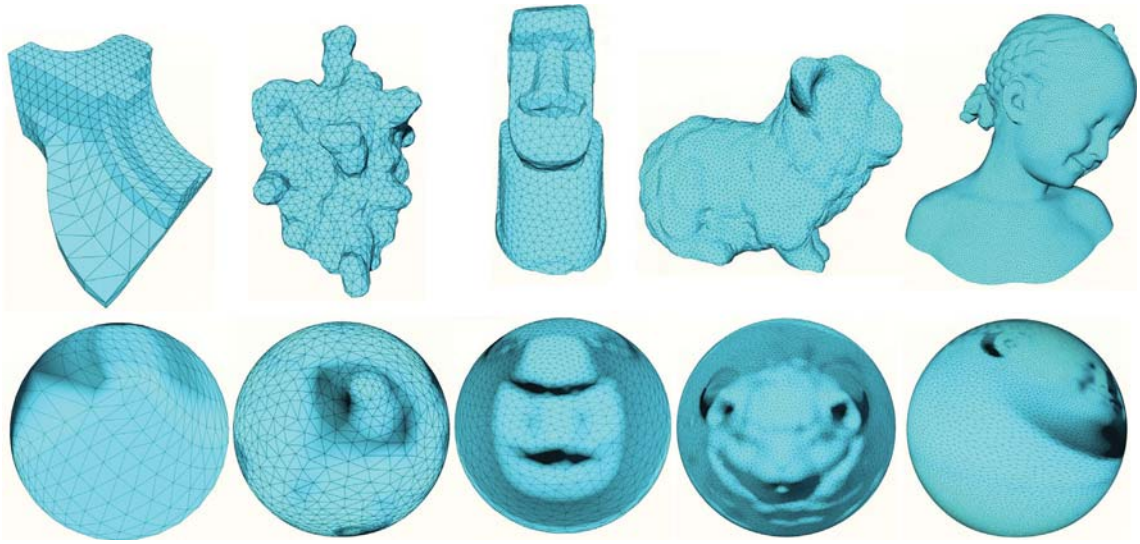

Fig. 10. Some results of our ARAP spherical parametrization. Numbers in bracket denote area, angle and rigidity distortion respectively. The r denotes the optimal radius.

Table 2
Comparison of area and angle distortion measures of different parametrization methods.

Model	D^{area}/D^{angle}				
	Convex	Harmonic	Conformal	Hierarchical	Ours
Head	3.0395/2.1785	2.5205/2.0573	2.5240/2.0036	2.0026/2.0062	2.0083/2.0026
Cube	9.1832/2.3248	2.5689/2.0457	2.5372/2.0097	2.0100/2.0763	2.0205/2.0176
David	2.8269/2.0247	2.6442/2.0591	2.5372/2.0124	2.0054/2.2095	2.1102/2.0476
Max-Planck	/	2.6056/2.3778	2.5688/2.0241	2.0564/2.3979	2.1824/2.0805
Fandisk	17.2302/2.5557	4.7324/2.5633	4.5619/2.0521	2.0277/2.5297	2.2455/2.1915
Blob	/	7.3849/2.3256	6.2833/2.0383	2.0479/2.5895	2.6174/2.5806
FilledPF	/	3.7538/2.5272	3.4528/2.0204	2.0517/2.5078	2.3449/2.1802
Moai	/	2.8513/2.4110	3.5531/2.0037	2.0191/2.5045	2.2945/2.1446
Bimba	/	5.8350/4.0238	5.1264/2.1011	2.1059/3.5362	2.4175/2.2817

Table 3
Comparison of rigidity distortion measures of different parametrization methods. The last column are the worst cases of the rigidity distortion measures.

Model	$D^{rigidity}$					
	Convex	Harmonic	Conformal	Hierarchical	Ours	Ours (worst case)
Head	0.5313	0.0610	0.0326	0.0177	0.0074	0.0901
Cube	0.9015	0.0702	0.0313	0.0418	0.0160	0.2444
David	0.2651	0.1647	0.1479	0.1101	0.0752	0.2613
Max-Planck	/	0.2305	0.1288	0.2320	0.0947	0.8272
Fandisk	0.9572	0.4147	0.2976	0.2847	0.1426	0.8500
Blob	/	0.4509	0.3496	0.3238	0.1845	1.0348
FilledPF	/	0.3440	0.2731	0.3081	0.1767	4.9937
Moai	/	0.4089	0.3962	0.2968	0.2148	1.1518
Bimba	/	0.8131	0.7399	1.1059	0.5111	11.3660

- Our approach has a small, even insignificant penalty in angle distortion compared with the conformal approach [11], but has a better value than other approaches [5–7].
- Our approach has a small, even insignificant penalty in area distortion compared with the hierarchical approach [7], but has a better value than other approaches [5,11,6].

From the tables, we find that our ARAP parametrization results have the best rigidity-preserving property compared with other methods. The area, angle and rigidity distortion are very low in all the examples of this paper, which demonstrates the efficiency and accuracy of our algorithm. We also provide the worst cases of the rigidity distortion measures in Table 3 for comparing with the average ones.

9. Conclusion

We have presented an efficient approach to parameterize the genus-zero triangular mesh onto the sphere with an optimal radius by minimizing the discrete ARAP energy. This is a natural extension from planar parametrization to spherical domain, corresponding to the extension of ARAP energy. However, as the extension involves a transition from non-constrained optimization to a highly nonlinear constrained one, it is much difficult to solve. We have provided a simple and efficient two-step iterative algorithm to solve the spherical parametrization problem. In the first step, we apply a local/global scheme to calculate the spherical parametrization coordinates. The local component tries to minimize the distance between the parametrization differential and the nearest rotation for each tetrahedron which individual triangle is in, while the global component fits all the tetrahedrons together. In second step, we try to find an optimal radius of the sphere for isometric mapping. We demonstrate our parametrization results on a collection of challenging models and show that our algorithm is simple, efficient and practical.

In future work, we will focus on the following problems. Since our objective energy cannot guarantee the parametrization results without flips especially for high curvature sections, we want to give out a better objective function to avoid the self-intersection, such as harmonic energy. In addition, we want to generalize our method to parameterize higher genus meshes onto sphere.

Acknowledgments

Thanks to Xin Lin and Ying Li for providing results of the algorithms of [7,6]. This work is supported by the National Natural Science Foundation of China (61222206) and the One Hundred Talent Project of the Chinese Academy of Sciences.

References

[1] A. Sheffer, E. Praun, K. Rose, Mesh parameterization methods and their applications, Found. Trends. Comput. Graph. Vis. 2 (2006) 105–171.
[2] L. Liu, L. Zhang, Y. Xu, C. Gotsman, S.J. Gortler, A local/global approach to mesh parameterization, in: Proceedings of the Symposium on Geometry Processing, SGP '08, 2008, pp. 1495–1504.
[3] O. Sorkine, M. Alexa, As-rigid-as-possible surface modeling, in: Proceedings of the Fifth Eurographics Symposium on Geometry Processing, SGP '07, 2007, pp. 109–116.
[4] X. Gu, Y. Wang, T.F. Chan, P.M. Thompson, S.-T. Yau, Genus zero surface conformal mapping and its application to brain surface mapping, IEEE Trans. Med. Imag. 23 (2004) 949–958.
[5] C. Gotsman, X. Gu, A. Sheffer, Fundamentals of spherical parameterization for 3d meshes, ACM Trans. Graph. (Proc. SIGGRAPH) 22 (2003) 358–363.
[6] Y. Li, Z. Yang, J. Deng, Spherical parametrization of genus-zero meshes by minimizing discrete harmonic energy, J. Zhejiang Univ. SCIENCE A 7 (2006) 1589–1595.
[7] S. Wan, T. Ye, M. Li, H. Zhang, X. Li, An efficient spherical mapping algorithm and its application on spherical harmonics, Sci. China Inform. Sci. 56 (2013) 1–10.
[8] M. Floater, K. Hormann, Surface parameterization: a tutorial and survey, in: Advances in Multiresolution for Geometric Modelling, Mathematics and Visualization, 2005, pp. 157–186.
[9] S. Haker, S. Angenent, A. Tannenbaum, R. Kikinis, G. Sapiro, M. Halle, Conformal surface parameterization for texture mapping, IEEE Trans. Visual. Comput. Graph. 6 (2000) 181–189.
[10] S. Saba, I. Yavneh, C. Gotsman, A. Sheffer, Practical spherical embedding of manifold triangle meshes, in: 2005 International Conference on Shape Modeling and Applications, 2005, pp. 256–265.
[11] X. Gu, S.-T. Yau, Global conformal surface parameterization, in: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '03, 2003, pp. 127–137.
[12] M. Jin, Y. Wang, S.-T. Yau, X. Gu, Optimal global conformal surface parameterization, in: Proceedings of the Conference on Visualization '04, VIS '04, 2004, pp. 267–274.
[13] H. Li, R. Hartley, Conformal spherical representation of 3d genus-zero meshes, Pattern Recogn. 40 (2007) 2742–2753.
[14] A. Sheffer, C. Gotsman, N. Dyn, Robust spherical parameterization of triangular meshes, Computing 72 (2004) 185–193.
[15] E. Praun, H. Hoppe, Spherical parametrization and remeshing, ACM Trans. Graph. (Proc. SIGGRAPH) 22 (2003) 340–349.
[16] P. Sander, J. Snyder, S. Gortler, H. Hoppe, Texture mapping progressive meshes, in: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01, 2001, pp. 409–416.
[17] R. Zayer, C. Rossl, H.-P. Seidel, Curvilinear spherical parameterization, in: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006, SMI '06, 2006, pp. 11–19.
[18] I. Friedel, P. Schrtöder, M. Desbrun, Unconstrained spherical parameterization, in: ACM SIGGRAPH 2005 Sketches, SIGGRAPH '05, 2005.
[19] C. Mantegazza, Lecture Notes on Mean Curvature Flow, Progress in Mathematics, Springer, Basel, 2011.
[20] G. Dziuk, An algorithm for evolutionary surfaces, Numer. Math. 58 (1990) 603–611.
[21] M. Kazhdan, J. Solomon, M. Ben-Chen, Can mean-curvature flow be modified to be non-singular?, Comput Graph. Forum 31 (2012) 1745–1754.
[22] B. Chow, L. Feng, Combinatorial ricci flows on surfaces, J. Differ. Geometry 63 (2003) 97–129.
[23] M. Jin, J. Kim, X. Gu, Discrete surface ricci flow: theory and applications, Math. Surf. XII 4647 (2007) 209–232.
[24] M. Jin, J. Kim, F. Luo, X. Gu, Discrete surface ricci flow, IEEE Trans.Visual. Comput. Graph. 14 (2008) 1030–1043.
[25] X. Chen, H. He, G. Zou, X. Zhang, X. Gu, J. Hua, Ricci flow-based spherical parameterization and surface registration, Comput. Vis. Image Understand. 117 (2013) 1107–1118.
[26] M. Alexa, D. Cohen-Or, D. Levin, As-rigid-as-possible shape interpolation, ACM Trans. Graph. (Proc. SIGGRAPH) (2000) 157–164.
[27] D. Xu, H. Zhang, Q. Wang, H. Bao, Poisson shape interpolation, in: Proceedings of the 2005 ACM Symposium on Solid and Physical Modeling, SPM '05, 2005, pp. 267–274.
[28] I. Chao, U. Pinkall, P. Sanan, P. Schröder, A simple geometric model for elastic deformations, ACM Trans. Graph. (Proc. SIGGRAPH) 29 (2010) 38:1–38:6.
[29] A. Myles, D. Zorin, Global parametrization by incremental flattening, ACM Trans. Graph. (Proc. SIGGRAPH) 31 (2012) 109:1–109:11.
[30] U. Pinkall, S.D. Juni, K. Polthier, Computing discrete minimal surfaces and their conjugates, Exp. Math. 2 (1993) 15–36.
[31] M. Meyer, M. Desbrun, P. Schröder, A.H. Barr, Discrete differential-geometry operators for triangulated 2-manifolds, Visual. Math. 3 (2002) 52–58.
[32] P. Degener, J. Meseth, R. Klein, An adaptable surface parameterization method, in: Proceedings of the 12th International Meshing Roundtable, 2003, pp. 201–213.
[33] K. Hormann, G. Greiner, MIPS: An efficient global parametrization method, in: Curve and Surface Design: Saint-Malo 1999, Innovations in Applied Mathematics, 2000, pp. 153–162.