Weighted area/angle distortion minimization for Mesh Parameterization

Daniel Mejia *(Laboratorio de CAD CAM CAE, Universidad EAFIT, Medellín, Colombia)*
Diego A. Acosta *(Grupo de Desarrollo y Diseño de Procesos, Universidad EAFIT, Medellín, Colombia)*
Oscar Ruiz-Salguero *(Laboratorio de CAD CAM CAE, Universidad EAFIT, Medellín, Colombia)*

Engineering Computations
ISSN: 0264-4401
Publication date: 7 August 2017
Reprints & Permissions

Weighted Area/Angle Distortion Minimization for Mesh Parameterization

Journal:	*Engineering Computations*
Manuscript ID	EC-02-2016-0072.R2
Manuscript Type:	Research Article
Keywords:	Reverse Engineering, Mesh Parameterization, Nonlinear Optimization, Levenberg-Marquardt, Complexity Analysis, Sensitivity Analysis

SCHOLARONE™
Manuscripts

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Weighted Area/Angle Distortion Minimization for Mesh Parameterization

Structured Abstract:

Purpose: Mesh Parameterization is central to reverse engineering, tool path planning, etc. This work synthesizes parameterizations with (a) un-constrained borders, (b) overall minimum angle plus (c) area distortion. We present an assessment of the sensitivity of the minimized distortion with respect to weighed area and angle distortions.

Methodology: A mesh parameterization is implemented which does not constrain borders by performing: (1) isometry maps for each triangle to the plane Z=0, (2) An affine transform within the plane Z=0 to glue the triangles back together and (3) a Levenberg-Marquardt minimization algorithm of a nonlinear F penalty function that modifies the parameters of the transformations (1) and (2) to discourage triangle flips, angle or area distortions. F is a convex weighed combination of area distortion (weight: α with $0 \leq \alpha \leq 1$) and angle distortion (weight: $1-\alpha$).

Value: (1) The devised free boundary nonlinear mesh parameterization method does not require a valid initial parameterization and produces locally bijective parameterizations in all of our tests. (2) A formal sensitivity analysis shows that the resulting parameterization is more stable (i.e. the *UV* mapping changes very little) when the algorithm tries to preserve angles than when it tries to preserve areas. (3) Our algorithm belongs to the class that parameterizes meshes with holes. (4) We present the results of a complexity analysis comparing our algorithm with 12 competing ones.

Findings: Our parameterization algorithm has linear complexity ($\mathcal{O}(n)$, $n =$ number of mesh vertices). The sensitivity analysis permits a fine-tuning of the weight parameter (α) which achieves overall bijective parameterizations in the studied cases. No theoretical guarantee is given in this manuscript for the bijectivity. Our algorithm has equal or superior performance compared with the ABF, LSCM and ARAP algorithms for the *Ball*, *Cow* and *Gargoyle* datasets. Additional correct results of our algorithm alone are presented for the *Foot*, *Fandisk* and *Sliced-Glove* datasets.

Keywords: Reverse Engineering, Mesh Parameterization, Nonlinear Optimization, Levenberg-Marquardt, Complexity Analysis, Sensitivity Analysis.

Article Classification: Research paper.

Abbreviations:

LM:	Levenberg-Marquardt.
I_k:	Identity matrix of degree k.
$\mathcal{O}(f(n))$:	Computational *time* complexity of an algorithm being asymptotic to $f(n)$, with n being the measuring unit.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

M:	Triangular mesh (with non-empty border) of a 2-manifold embedded in $\mathbb{R}^3$, composed by the set of triangles $T = \{t_1, t_2, \ldots, t_m\}$ with vertex set $X = \{x_1, x_2, \ldots, x_n\}$ ($X \subset \mathbb{R}^3$).
$\phi(x)$:	Parameterization of M which is a piecewise affine mapping $\{\phi_1, \phi_2, \ldots, \phi_m\}$ with $\phi_i = \psi_i \circ \eta_i$. $\phi: M \to \phi(M) \subset \mathbb{R}^2$ is a homeomorphism.
U:	Set of points $U = \{u_1, u_2, \ldots, u_n\}$ corresponding to the image of X under the mapping ϕ (i.e., $u_i = \phi(x_i)$).
η_i:	Rigid transformation which maps the triangle t_i of M to the plane Z=0 (also called XY plane here) and its center of mass $\bar{x}_i$ to the origin.
R^i:	Image of the vertices $[x_{i_1}, x_{i_2}, x_{i_3}]$ of the i-th triangle of M under the mapping η_i.
Q^i:	Right pseudo-inverse of R^i (i.e., $R^i Q^i = I_2$).
ψ_i:	An affine mapping $\mathbb{R}^2 \to \mathbb{R}^2$ which maps R^i to its final place in the parameterization (ϕ). $\psi_i(\xi) = A^i \xi + c_i$.
A^i:	Jacobian matrix of ψ_i.
D^i_{area}:	Area distortion of the triangle t_i under the mapping ϕ_i defined as $D^i_{area} = \left(\det(A^i) - 1\right)^2$.
D^i_{angle}:	Angle distortion of the triangle t_i under the mapping ϕ_i defined as $D^i_{angle} = (A^2_{11} - A^2_{22})^2 + (A^2_{12} - A^2_{21})^2$.
$F(U)$:	Penalty function $\mathbb{R}^{2n} \to \mathbb{R}$ to be minimized which sums the weighted area and angle distortion of the triangles in M under the mapping $U = \phi(X)$.
F^*:	Value at which the penalty function F is a local minimum.
α:	Parameter $0 \le \alpha \le 1$ which weighs area distortion (α) against angle distortion ($1 - \alpha$) in the penalty function F.
∇:	Gradient operator.
$\mathcal{H}$:	Hessian operator.
λ:	Damping parameter of the LM algorithm.
ε:	Tolerance parameter of the LM algorithm.
S^f_p:	Relative sensitivity of a penalty function f with respect to a p parameter.

1. Introduction

In CAD CAM CAE, it is usual to have a triangular mesh $M \subset \mathbb{R}^3$ as a result of the segmentation of a larger triangular mesh. M is a 2-manifold with non-empty border and low curvature (i.e., M is near-developable). Therefore, M admits a 2-variable parameterization which is a homeomorphism between M and a polygonal region in $\mathbb{R}^2$.

Mesh Parameterization consists of finding a mapping $\phi: M \to \phi(M) \subset \mathbb{R}^2$ such that: 1) ϕ and ϕ^{-1} are continuous (i.e., connectivity of the triangles is preserved after the mapping) and 2) ϕ is bijective (i.e., triangles do not overlap after the mapping). ϕ is a homeomorphism and the image of M under

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

ϕ is a parameterization of M. In addition, local preservation of properties (e.g., angle, area, dimensions, etc.) is pursued but rarely achieved in parameterizations of actual engineering M meshes.

Mesh Parameterization is relevant in areas such as reverse engineering, tool path planning, feature detection, re-design, etc. This article proposes an algorithm for computing ϕ by minimizing a penalty function F which discourages triangle flips, angle and area distortions. Different results may be obtained for the same mesh by changing the parameter α (which weighs angle vs. area preservation) in F, allowing the user to pick up the best parameterization. Fine-tuning of this parameter allows in some cases to reach globally bijective parameterizations from non-bijective ones.

The remainder of this article is structured as follows: Section 2 reviews the relevant literature. Section 3 describes the implemented methodology. Section 4 presents and discusses the results of the test runs. Section 5 concludes the paper and introduces opportunities for future work.

2. Literature Review

Mesh Parameterization is usually achieved by posing an optimization problem where some kind of distortion measure is minimized in the parameter space. Depending on the characteristics of the method, Mesh Parameterization algorithms can be classified into: 1) Constrained-Boundary methods, 2) Free-Boundary methods, or 3) Dimensionality Reduction (DR) methods. Refs. (Hormann *et al.*, 2007; Sheffer *et al.*, 2006) present a survey of the state of the art in Mesh Parameterization methods. Extending such surveys, this section discusses the relevant literature in the topic.

2.1. Constrained-Boundary Mesh Parameterization

In Constrained-Boundary parameterizations, the border of the mesh is constrained in the resulting parameterization. Such constraint is forced by mapping the vertices of the mesh border to a fixed shape (e.g., a disk) in the parameter space. Barycentric coordinates methods (Floater, 1997) solve the parameterization problem by expressing each vertex in the parameter space as a convex combination of its neighbors. In Ref. (Yoshizawa *et al.*, 2004), a nonlinear-gradient algorithm which minimizes a stretch measure on a given Floater parameterization is proposed. The Signal-Specialized method (Sander *et al.*, 2002) proposes a non-linear algorithm for mapping the surface to a rectangular domain based on a function defined on the surface (such as a color map). In Ref. (Pietroni *et al.*, 2011), a local flattening operator is proposed to interactively parameterize rectangular patches on a surface to their corresponding planar domain. In Ref. (Zou *et al.*, 2011), an area-preserving mapping is sought by simulation of Lie advection (moving mass property change) on the surface, while in Refs. (Zhao *et al.*, 2013; Su *et al.*, 2016) area-preserving mappings are computed using the optimal mass transport technique.

Constrained-Boundary methods present an additional problem of mapping the boundary to the parameter space in a separate step. This problem leads to highly distorted mappings in most application cases. The Virtual Boundary algorithm (Lee *et al.*, 2002) partially overcomes this problem by introducing an artificial boundary connected to the real boundary. However, the shape of the artificial boundary affects the result and still introduces several distortions.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

2.2. Free-Boundary Mesh Parameterization

In contrast to Constrained-Boundary parameterizations, Free-Boundary methods do not require fixing the boundary in the parameter space. The MIPS (Most Isometric Parameterizations) method (Hormann and Greiner, 2000) proposes to minimize a nonlinear-gradient Dirichlet energy without boundary conditions, which produces a free boundary parameterization. Similarly, the Intrinsic Parameterizations algorithm (Desbrun *et al.*, 2002), minimizes an energy functional which is a linear combination of both area and angle distortions. LSCM (Least Squares Conformal Maps) seeks a conformal parameterization by discretizing the Cauchy-Riemann equations on a triangular mesh (Lévy *et al.*, 2002). The One-Step Inverse Forming approach (Li *et al.*, 2010; Zhu *et al.*, 2013) is a mesh parameterization algorithm based on the simulation of physical plastic deformation of the surface. In Ref. (Li *et al.*, 2012), an isometric parameterization is sought by computing geodesics on the surface. The aforementioned Free-Boundary algorithms present the shortcoming of only accepting surfaces that are homeomorphic to a disk in $\mathbb{R}^2$ (i.e. surfaces without holes).

The ABF (Angle-Based Flattening) algorithm (Sheffer and de Sturler, 2001) is a nonlinear-gradient algorithm which poses an optimization problem in terms of the angles of the mesh triangles in order to find a conformal parameterization. However, the ABF algorithm is computationally expensive making it impractical for large datasets. The ABF++ algorithm (Sheffer *et al.*, 2005) and the Linear ABF (Zayer *et al.*, 2007) present variations to the original ABF algorithm which potentially improves the computation time at the cost of global distortion. Angle-based algorithms usually require a post-processing step where the parameterization must be recovered from the computed angles of the mesh. The Circle Patterns (Kharevych *et al.*, 2006), Curvature Prescription (Ben-Chen *et al.*, 2008), Conformal Equivalence (Springborn *et al.*, 2008) and Controlled-distortion (Myles and Zorin, 2013) algorithms seek conformal parameterizations by transferring the Gaussian curvature of the triangular mesh to a selected set of nodes known as *cone singularities*. A similar approach extended to hexagonal meshes is proposed in Ref. (Nieser *et al.*, 2012). Automatic algorithms for placing such *cone singularities* on the mesh have been proposed in the literature (Ben-Chen *et al.*, 2008; Springborn *et al.*, 2008; Myles and Zorin, 2012). However, such placement is an additional pre-processing step to the Mesh Parameterization process that increases the complexity of the algorithm.

In Ref. (Zayer *et al.*, 2005), a nonlinear-gradient algorithm which uses conformal and quasi-harmonic maps for parameterizing surfaces with holes is proposed. ASAP/ARAP (As Similar As Possible / As Rigid As Possible) and ARAP++ (Liu *et al.*, 2008; Wang *et al.*, 2016) minimize an energy functional which is a linear combination of both area and angle distortions. However, the ASAP/ARAP method requires a post-processing step due to triangle flips occurring in the resulting parameterization. The Constrained Parameterization on Parallel Platforms algorithm (Athanasiadis *et al.*, 2013) minimizes an energy function similar to the MIPS energy, using GPU resources to improve the computational efficiency. These nonlinear-gradient methods allow to parameterize surfaces with holes as opposed to previous methods. However, nonlinear-gradient methods require computing an initial valid parameterization (i.e., with no triangle flips) which increases the computational cost of the algorithm and the implementation complexity.

The problem of globally bijective parameterizations cannot be guaranteed in most cases (especially for Free-Boundary methods). The recently proposed Bijective Parameterizations with Free Boundaries algorithm (Smith and Schaefer, 2015) overcomes this problem by posing a nonlinear optimization problem with barrier functions which do not allow boundary overlapping. However, this algorithm becomes highly expensive when evaluating boundary overlaps. In addition, as most nonlinear-gradient parameterization algorithms it requires an initial valid parameterization.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Engineering Computations

2.3. *Dimensionality Reduction for Mesh Parameterization*

Mesh Parameterization is an application of Dimensionality Reduction (DR). DR techniques perform a parameterization of a k-manifold M ($k \in \mathbb{N}$), using the information about a proximity graph of the manifold. DR does not, in general, take advantage of the explicit mesh triangulation structure. Algorithms such as Isomap (Tenenbaum *et al.*, 2000), Laplacian Eigenmaps (Belkin and Niyogi, 2003), LTSA (Local Tangent Space Alignment) (Zhang and Zha, 2002), LLE (Locally Linear Embedding) (Roweis and Saul, 2000) and HLLE (Hessian Locally Linear Embedding) (Donoho and Grimes, 2003) are popular DR algorithms which have been relevant in the mesh parameterization literature. In Ref. (Sun and Hancock, 2008), a method that combines Isomap and barycentric coordinates is proposed in order to produce an isometric parameterization. However, such method relies on estimation of geodesics which is inappropriate for non-convex manifolds. In addition, geodesics are computationally expensive to estimate. Refs. (Ruiz *et al.*, 2015) (using Laplacian Eigenmaps) and (Mejia *et al.*, 2016) (using a modified HLLE) avoid the computation of geodesics. This modification largely offsets the problems related to geodesic distance estimation. However, they cannot guarantee preservation of angle or areas resulting in high distortions. The Optimal Local Flattening algorithm (Chen *et al.*, 2007) is a Mesh Parameterization algorithm based on the LTSA dimensionality reduction technique which presents low distortions. However, triangle flips (local non-bijectivity) may arise during the mapping.

2.4. *Conclusions of the literature review*

As discussed earlier, current Mesh Parameterization algorithms may present one or more of the following disadvantages: constrained boundary, local overlaps (triangle flips), requirement of an initial valid parameterization to avoid local minima (for the penalty function) and non-bijective mappings, etc. In this article we propose a free-boundary mesh parameterization algorithm where each triangle is mapped individually to the plane *Z=0* by a rigid transformation and then a penalty function F measuring distortion in the global parameterization is minimized. In contrast to most non-linear gradient algorithms, our algorithm does not require an initial valid parameterization. We observed no triangle flips in the cases processed by our algorithm, although we have no theoretical guarantee for such a behavior. A $0 \leq \alpha \leq 1$ parameter weighs area distortion against angle distortion weighed by $1 - \alpha$. The weighting scheme proposed in this manuscript restricts the α parameter by a convex combination which potentially avoids numerical instabilities that may arise as opposed to unbounded weighting parameters as in Refs. (Desbrun *et al.*, 2002; Degener *et al.*, 2003; Liu *et al.*, 2008).

In order to minimize F we implement the Levenberg-Marquardt (LM) algorithm. LM is a gradient descent method with high convergence rate allowing to evade the computation of an initial valid parameterization which is a requirement in most nonlinear-gradient algorithms. The test runs consider both surfaces with and without holes, showing that no triangle flips occur in the resulting parameterization. A fine-tuning of the α parameter results in globally bijective parameterizations. We present a complexity analysis of our algorithm and a sensitivity analysis for the minimized F^* with respect to α. As per our Literature Review, a sensitivity analysis for weighting parameters has not been yet presented in the Mesh Parameterization.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Engineering Computations

3. Methodology

Consider $M=(X,T)$ a connected 2-manifold in $\mathbb{R}^3$ with border which is homeomorphic to a polygonal region in $\mathbb{R}^2$, possibly with holes. Our goal is to find a set of points $U=\{u_1,u_2,\dots,u_n\}\subset\mathbb{R}^2$ such that u_i is the image of $x_i\in X$ under a homeomorphism $\phi: M\rightarrow\mathbb{R}^2$.

We build ϕ as the composition $\psi\circ\eta$, as follows: (1) $\eta: M\rightarrow XY\subset\mathbb{R}^3$ maps in arbitrary rigid manner each triangle t_i of M onto the plane Z=0. (2) An affine mapping ψ in $\mathbb{R}^2$ glues the 2D (parametric space) triangles as they originally were in M. The function ϕ presents a compromise between angle vs. area preservation. The extent of such a compromise is part of the findings of this manuscript.

The algorithm for finding the parameterization $U=\phi(X)$ is described below (Fig. 1):

a. **Rigid mapping $\boldsymbol{\eta_i}: \boldsymbol{M\rightarrow XY}\subset\mathbb{R}^3$:** Find the rigid transformation $\eta_i: M\rightarrow XY\subset\mathbb{R}^3$ that maps the triangle t_i to the XY plane and its center of mass $\bar{x}_i$ to the origin. The matrix $R^i=\eta([x_{i_1},x_{i_2},x_{i_3}])$ corresponds therefore to the image of the vertices $[x_{i_1},x_{i_2},x_{i_3}]$ of the triangle t_i under such mapping.

b. **Affine mapping $\boldsymbol{\psi_i}: \mathbb{R}^2\rightarrow\mathbb{R}^2$:** Since each triangle has been mapped individually to $\mathbb{R}^2$, an affine mapping $\psi_i(\xi)=A^i\xi+c_i$ which maps each R^i to the final parameterization ϕ is constructed. The Jacobian matrix A^i can be computed in terms of R^i and the vertices $[u_{i_1},u_{i_2},u_{i_3}]$ of the triangle $\phi(t_i)$. From this construction, $\phi_i=\psi_i\circ\eta_i$ is an affine mapping and $\phi=\{\phi_1,\phi_2,\dots,\phi_m\}$ is a piecewise affine mapping which parameterizes M. The continuity of ϕ is implied in $\psi=\{\psi_1,\psi_2,\dots,\psi_m\}$ from the connectivity of M, i.e., if t_i and t_j share the edge (x_k,x_l) then ψ_i and ψ_j overlap in the edge (u_k,u_l).

c. **Weighted penalty function $\boldsymbol{F(U)}$:** A penalty function $F:\mathbb{R}^{2n}\rightarrow\mathbb{R}$ which penalizes the weighted area and shape distortion of each triangle is constructed in this step. Since $\phi_i=\psi_i\circ\eta_i$ is an affine mapping and η_i is rigid, all the distortion of ϕ_i can be extracted from A^i. An area (D^i_{area}) and angle (D^i_{angle}) distortion is build for each triangle in terms of A^i, and a weighted sum of these terms over all the triangles compose the penalty function F.

d. **Parameterization $\boldsymbol{U=\phi(X)}$:** Since ϕ has a minimum distortion, U is estimated by minimizing F. Because ∇F is nonlinear, we implement the LM algorithm for this optimization process.

A detailed discussion follows.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

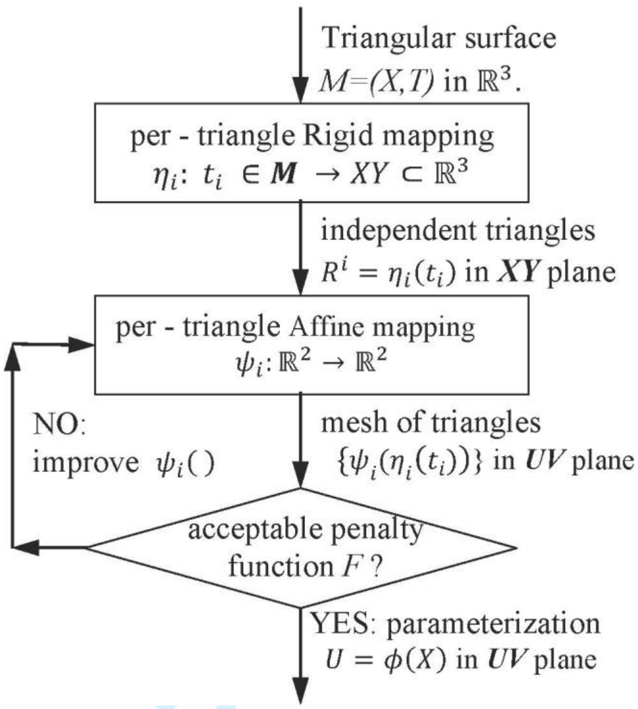


Figure 1: Implemented Mesh Parameterization algorithm.

3.1. *Rigid mapping* $\eta_i: M \to XY \subset \mathbb{R}^3$

In order to build the function F, we propose to map individually each triangle t_i of M to the XY plane first. One way to do this is to compute the center of mass $\bar{x}_i$ and the normal vector $\vec{n}_i$ of the triangle t_i. If $B_i = [\vec{v}_1^i, \vec{v}_2^i]$ is an orthonormal basis of the plane with normal $\vec{n}_i$, then $\eta_i: \mathbb{R}^3 \to XY \subset \mathbb{R}^3$ defined as:

$$\eta_i(x) = B_i^T(x - \bar{x}_i), \tag{1}$$

is a projection which maps isometrically the triangle t_i to the plane tangent to t_i (Fig. 2). Therefore, the matrix R^i corresponds to the image of the current triangle vertices under the map η_i, i.e.:

$$R^i = \eta_i([x_{i_1}, x_{i_2}, x_{i_3}]), \tag{2}$$

where x_{i_j} is the j-th vertex of t_i.

(a) A triangle (t_i) on the surface M.

(b) Mapping of the triangle t_i to the plane XY. The mapped triangle R^i is isometric to t_i and it is mean centered.

Figure 2: Mapping of a triangle on M to $\mathbb{R}^2$ by projecting it onto the tangent plane.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Engineering Computations
Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

3.2. *Affine mapping* $\psi_i: \mathbb{R}^2 \to \mathbb{R}^2$

ϕ_i is an affine transformation such that $\phi_i = \psi_i \circ \eta_i$, where $\psi_i: \mathbb{R}^2 \to \mathbb{R}^2$ is an affine transformation that maps R^i to the global parameterization U. Therefore:

$$\psi_i(\xi) = A^i \xi + c_i, \tag{3}$$

where A^i is a 2×2 linear transformation and $c_i = \frac{1}{3}\sum_{j=1}^{3} u_{i_j}$ is a translation term corresponding to the center of mass of the triangle $\phi(t_i)$.

Since η_i is isometric to t_i, the matrix A^i contains all the information about the distortion of the triangle t_i under the mapping ϕ_i. Recalling that $\phi(t_i) = \psi_i(R^i)$, we solve Eq. (3) for A^i:

$$A^i = [u_{i_1} - c_i, u_{i_2} - c_i, u_{i_3} - c_i]Q^i, \tag{4}$$

where Q^i is the right pseudoinverse of R^i (i.e., $R^i Q^i = I_2$). The preservation of the connectivity of M under ϕ is implied in Eq. (4). Specifically, the set of matrices $\boldsymbol{A} = \{A^1, A^2, \ldots, A^m\}$ are correlated in the sense that if t_i and t_j share an edge (x_k, x_l), the matrices A^i and A^j share the terms u_k, u_l.

3.3. *Weighted penalty function* $F(U)$

Calculating A^i in terms of the parameterization coordinates U allows to evaluate the local authalic (area) and conformal (angle) distortion for each triangle under the parameterization ϕ. A transformation is authalic if and only if its Jacobian determinant is ± 1. A consistent orientation is important to avoid local overlaps (triangle flips) which may result in a non-bijective mapping, therefore the minus sign is discarded. Thus we measure the area distortion D^i_{area} on each triangle by setting:

$$D^i_{area} = \left(\det(A^i) - 1\right)^2 \tag{5}$$

On the other hand, a mapping is conformal if its Jacobian matrix is k times a rotation matrix. Similar to (Lévy *et al.*, 2002), we construct the angle distortion measure D^i_{angle} of each triangle as follows:

$$D^i_{angle} = \left(A^i_{11} - A^i_{22}\right)^2 + \left(A^i_{12} + A^i_{21}\right)^2 \tag{6}$$

If we define the area and shape distortion of the mapping as the weighted sum of all D^i_{area} and all D^i_{angle} respectively, we can measure the global distortion as a convex combination of the global area and angle distortion:

$$F = \sum_{i=1}^{m} \alpha D^i_{area} + (1 - \alpha) D^i_{angle}, \tag{7}$$

with $0 \leq \alpha \leq 1$ being a weighting parameter such that F measures only angle distortion if $\alpha \to 0$ and F measures only area distortion if $\alpha \to 1$. F is only dependant of $\boldsymbol{Q} = \{Q^1, Q^2, \ldots, Q^m\}$ and $U = \{u_1, u_2, \ldots, u_n\}$ which are required to compute A^i as described in Eq. (4) and the weighting parameter α is introduced in order to control the resulting parameterization. The parameter α produces a compromise between area - preserving vs. angle - preserving parameterizations. This compromise is central in the cases were both criteria cannot be satisfied. Simultaneous area and angle preservation

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

is only possible with isometric parameterizations, which are feasible only for the special case of developable surfaces (e.g. with null Gaussian curvature at each surface point).

3.4. *Parameterization* $U = \phi(X)$

In order to find the global parameterization $U = \phi(X)$ it is necessary to minimize the F defined in Eq. (7). Therefore our global parameterization is given by the value of U that solves the following unconstrained problem:

$$\min_{U} F = \left\{ \sum_{i=1}^{m} \alpha D_{area}^{i} + (1-\alpha) D_{angle}^{i} \right\}, \tag{8}$$

The function F from Eq. (7) is continuous and has $2n$ degrees of freedom (2 degrees for each coordinate value u_i). The nonlinear nature of the gradient of F requires a nonlinear method for finding a solution to Eq. (8). Therefore, we choose the Levenberg-Marquardt (LM) algorithm for such a purpose, which is described below.

3.5. *The Levenberg-Marquardt (LM) algorithm*

We use a LM algorithm to update the parameterization coordinates according to the following scheme (Ravindran *et al.*, 2007):

$$U^{k+1} = U^{k} - \left(\mathcal{H}\left[F\left(U^{k}\right)\right] + \lambda^{k} I_{2n}\right)^{-1} \nabla F\left(U^{k}\right), \tag{9}$$

where $k+1$ is the current iteration, λ^k is the LM damping parameter which is updated iteratively according to the current solution, and $\mathcal{H}\left[F\left(U^{k}\right)\right]$ is the Hessian matrix of F defined as $\mathcal{H}_{ij}[F] = \frac{\partial^2 F}{\partial u_i \partial u_j}$ (Papadimitriou and Steiglitz, 1982).

We coded an implementation of the LM algorithm in MATLAB for the solution of Eq. (8). The advantages of our implementation are described below:

1. **Initial parameterization:** Our implementation allows an initial random parameterization U^0 for iterating Eq. (9), providing consistent parameterizations in all our test cases. This is superior to most nonlinear-gradient algorithms which require an initial valid parameterization (estimated by a linear parameterization algorithm) to proceed, such as in Refs. (Athanasiadis *et al.*, 2013; Liu *et al.*, 2008; Smith and Schaefer, 2015).
2. **Hessian estimator:** The LM algorithm proposes to estimate the Hessian matrix as $\mathcal{H}[F] \approx \nabla F \cdot \nabla F^T$. This approach leads to a dense Hessian matrix. Our implementation computes the Hessian of each triangle distortion ($\alpha \mathcal{H}\left[D_{area}^{i}\right] + (1-\alpha)\mathcal{H}\left[D_{angle}^{i}\right]$) individually, and then adds up such terms to the global Hessian matrix $\mathcal{H}[F]$. Therefore, our estimated Hessian $\mathcal{H}[F]$ ends up being a $2n \times 2n$ (with n = number of mesh nodes) symmetric sparse matrix where only adjacent points in M have respective nonzero elements. The sparsity of our Hessian matrix has positive effects (Sect. 4.5) on the computing expenses of our algorithm.

The iterative procedure is applied on Eq. (9) until certain criteria is met: 1) the norm of the gradient $\|\nabla F\|$ is lower than a fixed tolerance ε ($\varepsilon \in \mathbb{R}$) or 2) a certain number of iterations has occurred.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

3.6. *Sensitivity analysis*

We perform a relative sensitivity analysis of the minimized penalty function F^* with respect to the weighing parameter α to compare the resulting paramaterization and the global distortion for a chosen value for α numerically (Edgar and Himmelblau, 2001):

$$S_\alpha^F = \frac{\partial \ln F^*}{\partial \ln \alpha} = \frac{\alpha}{F^*}\frac{\partial F^*}{\partial \alpha} \approx \frac{\bar{\alpha}}{\overline{\overline{F^*}}}\frac{\Delta F^*}{\Delta \alpha} \quad (10)$$

Eq. (10) provides an idea of how small changes in the α parameter impact the minimized penalty function F^* given mesh M.

3.7. *Computer Experiment Set Up*

The tests run to assess our algorithm performance include several data sets and comparison with competitor algorithms. Table 1 discusses the data sets used. Table 2 lists the competitor algorithms tested along with ours. We do not intend to run a full benchmark test because we cannot guarantee even conditions to run all algorithms.

	Natural Border	In - source added boundary	Origin
Tests Run in Competition with ARAP, ABF, LSCM.			
Balls	Yes	None	ARAP site
Beetle	Yes	None	ARAP site
Cow	No (closed)	Yes (Sheffer and Hart, 2002)	Ref. (ALICE project-team, 2008)
Gargoyle	No (closed)	Yes (Sheffer and Hart, 2002)	ARAP site
Tests of our algorithm alone			
Bull	No (closed)	Yes (Sheffer and Hart, 2002)	Ref. (ALICE project-team, 2008)
Foot	Yes	Yes (Sheffer and Hart, 2002)	Ref. (ALICE project-team, 2008)
Fandisk	No (closed)	Yes (Sheffer and Hart, 2002)	Ref. (ALICE project-team, 2008)
Sliced - Glove	Yes.	Yes (open cut to use one hemisphere)	Ref. (ALICE project-team, 2008).

ARAP site: http://www.math.zju.edu.cn/ligangliu/cagd/Projects/ARAPPara/default.htm

Table 1: Datasets used for the algorithm appraisal.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

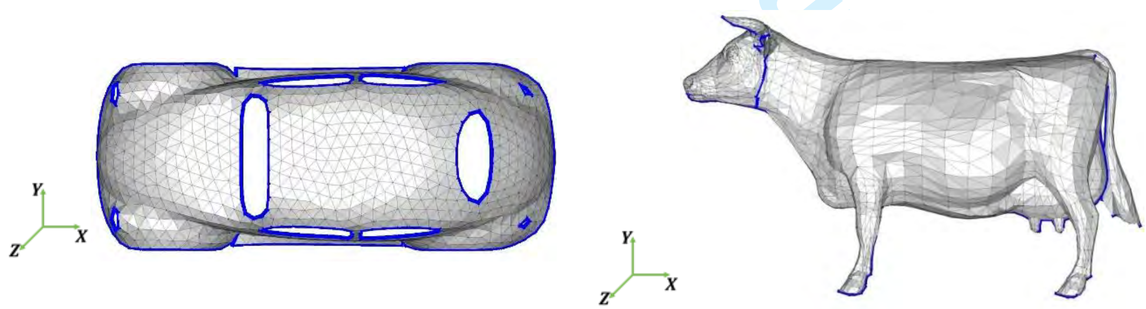
Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

	Requires a valid initial parameterization	Generality of tests	Reference	Code
ARAP	Yes	Restricted for data sets *Balls, Beetle, Gargoyle, Cow*	(Liu *et al.*, 2008)	Interpreted (MATLAB)
ABF	No	No restricted	(Sheffer and de Sturler, E. 2001)	Compiled (C)
LSCM	No	No restricted	(Lévy *et al.*, 2002)	Compiled (C)
Our algorithm	No	No restricted	This article.	Interpreted (MATLAB)

Table 2: Conditions of competitor algorithms considered.

4. Results and discussion

In this section, two case studies from the literature are presented and analyzed thoroughly. Section 4.1 presents the first case corresponding to the *Beetle* dataset (Fig. 3(a)). We show that by tuning adequately the α parameter, a valid parameterization can be achieved. Section 4.2 presents the second case study namely the *Cow* dataset (Fig. 3(b)). This case study has presented several problems and though a nearly-valid parameterization is achieved with our algorithm, global overlaps cannot be helped (Sheffer and de Sturler, 2001; Smith and Schaefer, 2015). Section 4.3 presents and discusses a summary of the results of our parameterization algorithm applied to other datasets and Sect. 4.4 compares our parameterization results with ABF, LSCM and ARAP. Finally, Sect. 4.5 presents the results of a complexity analysis comparing our algorithm with 12 competing ones.



(a) *Beetle* dataset. (b) *Cow* dataset.

Figure 3: Case studies datasets.

4.1. *Beetle dataset results*

Fig. 4 presents the resulting parameterization U for the *Beetle* dataset with different values of α. Setting $\alpha = 0.1$ results in a valid parameterization with low shape distortion as seen in Fig. 4(a). This

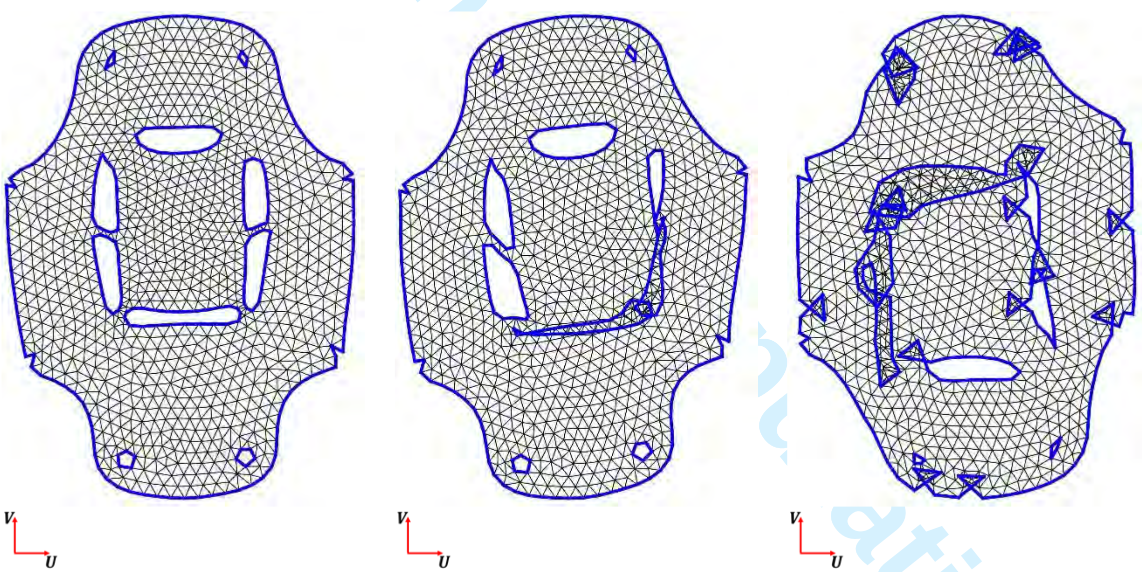
Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

is not the case for $\alpha = 0.5$ (Fig. 4(b)), where global overlaps occur as some area preservation is demanded to the algorithm (triangle flips do not happen). A highly authalic mapping ($\alpha = 0.9$) results in a parameterization with higher shape distortion and low area distortion (Fig.4(c)). Despite no triangle flips occur, the boundary and non-adjacent triangles overlap resulting in a non-bijective parameterization. The mapped texture in Fig. 5 shows how the shape is highly preserved as squares attain its form through the bijective mapping for $\alpha = 0.1$. Similar results for this dataset have been presented by other authors (Liu *et al.*, 2008; Sun and Hancock, 2008).

Recalling that the implemented algorithm converges to the same solution despite the initial (possibly non-valid) parameterization, Fig. 6 presents the initial, intermediate and final stages for $\alpha = 0.1$ of the LM for different initial parameterizations: i) an initial parameterization U^0_{Isomap} computed by the Isomap algorithm (Tenenbaum *et al.*, 2000) (Fig. 6(a)), ii) an initial parameterization U^0_{LapEig} computed by the Laplacian Eigenmaps algorithm (Belkin and Niyogi, 2003) (Fig. 6(b)) and iii) a randomly generated (non-bijective) initial parameterization U^0_{Rand} (Fig. 6(c)). We coded an implementation of both DR algorithms (Isomap and Laplacian Eigenmaps) in MATLAB, while the random parameterization is generated by the MATLAB *rand()* routine. The respective intermediate steps (Figs. 6(d), 6(e) and 6(f)) show how the surface is unfolded in each case and Fig. 6(g) presents the resulting parameterization for all the cases illustrating the consistency of the algorithm (even for the random non-valid initial parameterization U^0_{Rand}) as discussed in section 3.5.



(a) $\alpha = 0.1$. (b) $\alpha = 0.5$. (c) $\alpha = 0.9$.

Figure 4: Resulting parameterization for the *Beetle* dataset for different α values: (a) $\alpha = 0.1$ (quasi-conformal), (b) $\alpha = 0.5$ (conformal and authalic), (c) $\alpha = 0.9$ (quasi-authalic).

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

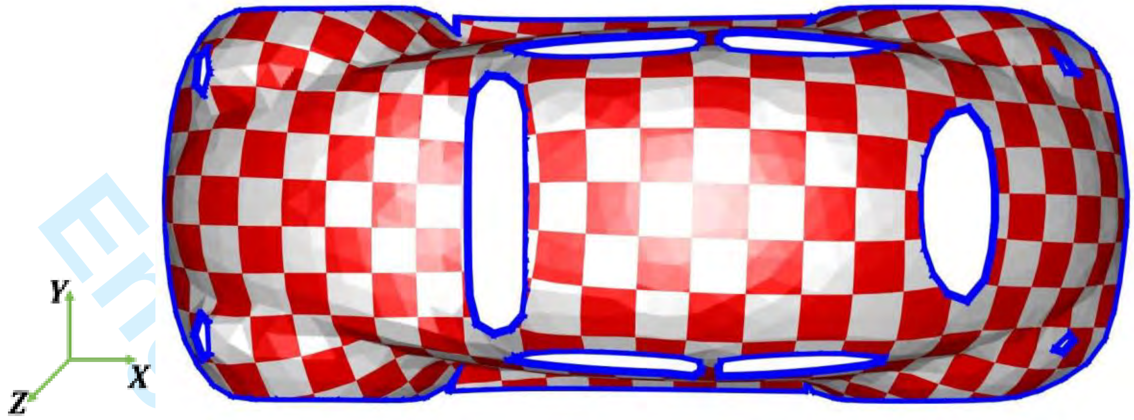
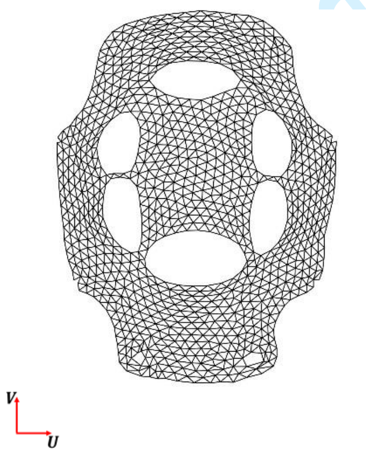
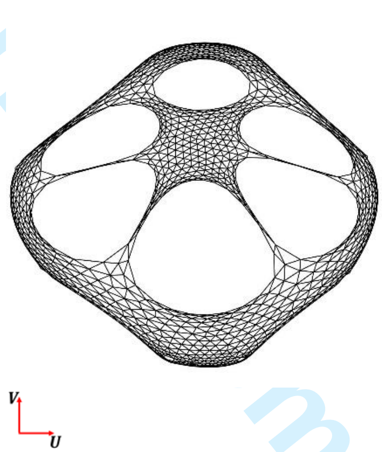


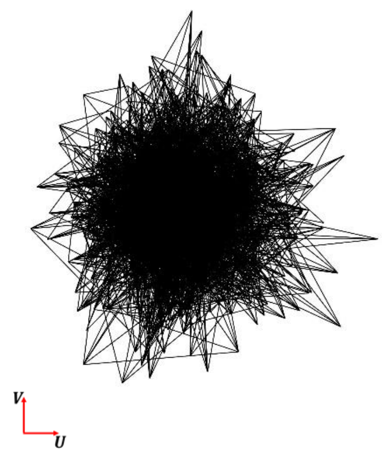
Figure 5: Texture map for the *Beetle* dataset ($\alpha = 0.1$).



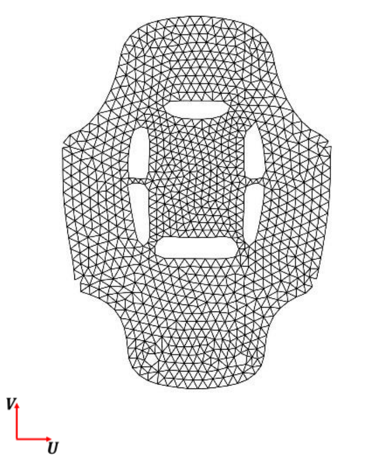
(a) Isomap initial parameterization U^0_{Isomap}.



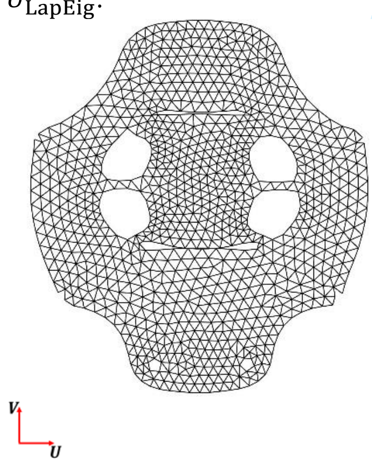
(b) Laplacian Eigenmaps initial parameterization U^0_{LapEig}.



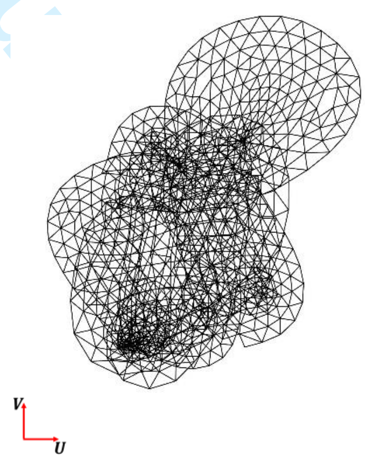
(c) Random (non-bijective) initial parameterization U^0_{Rand}.



(d) Intermediate solution for U^0_{Isomap}.



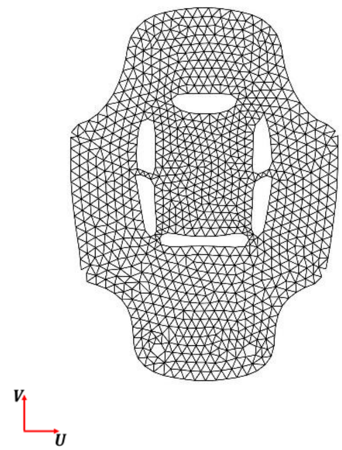
(e) Intermediate solution for U^0_{LapEig}.



(f) Intermediate solution for U^0_{Rand}.

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

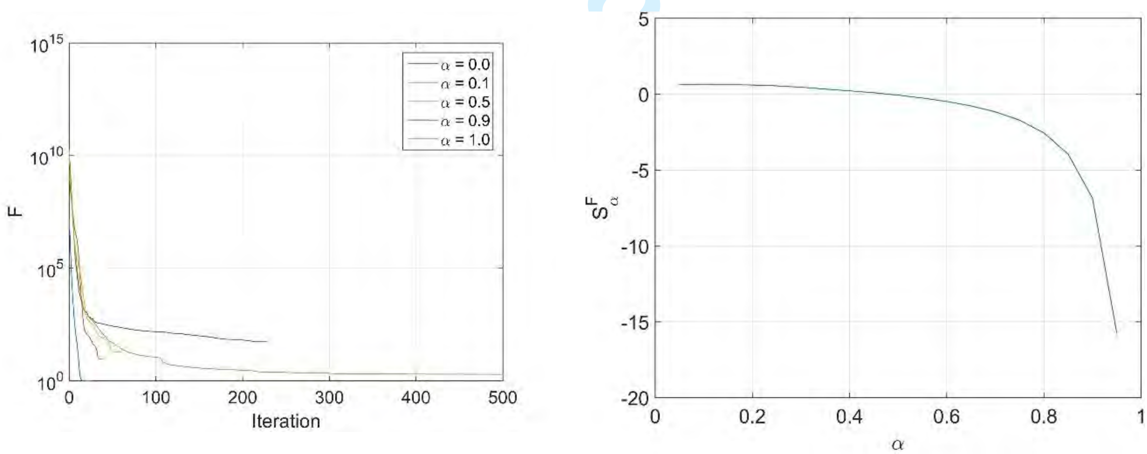
Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072



(g) Final LM parameterization for: i) U^0_{Isomap}, ii) U^0_{LapEig} and iii) U^0_{Rand}.

Figure 6: Initial, intermediate and final stages of LM optimization for the *Beetle* dataset using different initial parameterizations: i) Isomap U^0_{Isomap}, ii) Laplacian Eigenmaps U^0_{LapEig} and iii) random parameterization U^0_{Rand}. α is set to 0.1.

Using the random initial solution of Fig. 6(c), Fig. 7(a) presents the evolution of the penalty function F for different α values. Higher values of α take more iterations before converging. Also, though F^* reaches a lower value for $\alpha = 1$ than for $\alpha = 0.5$, this is not guarantee of a better result as seen in Figs. 4(b) and 4(c). Fig. 7(b) plots the relative sensitivity of F^* with respect to α as per Eq. (12). F^* becomes highly sensitive to α for values greater than 0.6. In this particular case, the resulting parameterization becomes non-valid for higher values of α as seen in Figs. 4(a) and 4(b).



(a) Evolution of F for different α values. (b) Relative sensitivity S^F_α.

Figure 7: *Beetle* dataset. Sensitivity analysis of F with respect to α. For area-preserving parameterizations, (a) the algorithm evidences slower convergence for higher values of α and (b) the penalty function F is highly sensitive in the area preserving ($\alpha \to 1$) side.

4.2. *Cow dataset results*

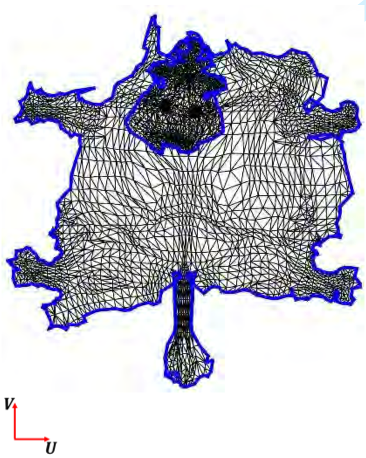
For a random initial parameterization U^0_{Rand}, Fig. 8 presents the resulting parameterization U for the *Cow* dataset with different α values. Setting $\alpha = 0.1$ results in a non-valid parameterization where

Engineering Computations
Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

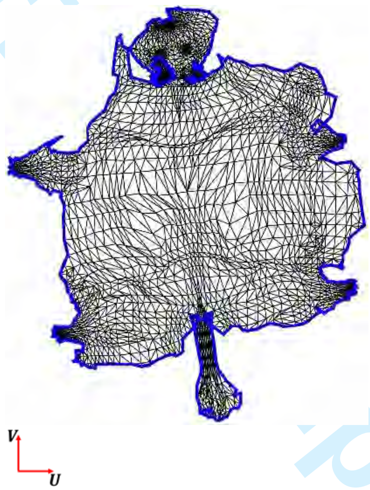
Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

the head of the *Cow* overlaps its body (Fig. 8(a)). For $\alpha = 0.01$, the head no longer overlaps the body in the resulting parameterization (Fig. 8(b)). However, the resulting parameterization is still non-bijective. Attempting a purely conformal parameterization ($\alpha = 0$) results in a high distorted mapping where the head and the legs present a high area distortion (Figs. 8(c) and 9). Zooming into the head and tail of the *Cow* it is clear that the boundary self-intersects and the parameterization is not bijective (Figs. 8(d) and 8(e)). It is important to emphasize that none of the discussed results above presents triangle flips despite the map not being bijective. Our results are in concordance with other authors (Ben-Chen *et al.*, 2008; Liu *et al.*, 2008; Sheffer *et al.*, 2005; Zayer *et al.* 2007) where non-bijective parameterizations (e.g., global overlaps) have been also reported for the *Cow* dataset. Finally, higher values of α were tested resulting in worse parameterizations.

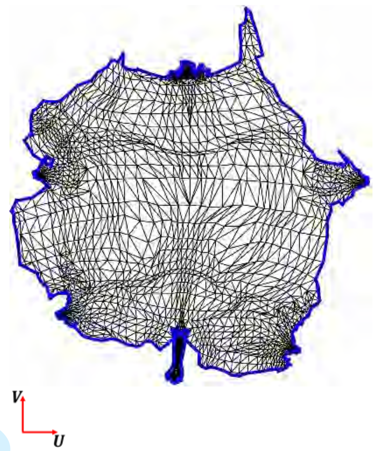
Fig. 10 presents a sensitivity analysis of F with respect to α for the *Cow* dataset. Again, higher values of α require the algorithm more iterations to converge (Fig. 10(a)). Similar to our *Beetle* sensitivity results, the relative sensitivity of F with respect to α (Fig. 10(b)) shows how F becomes highly sensitive for higher values of α ($\alpha > 0$). Again, the sensitivity analysis must be complemented by the user criteria as different values for α produce different (and not necessarily valid) parameterizations (Fig. 8).



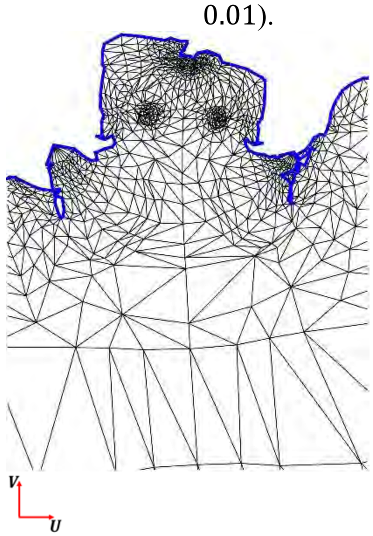
(a) *Cow* parameterization ($\alpha = 0.1$).

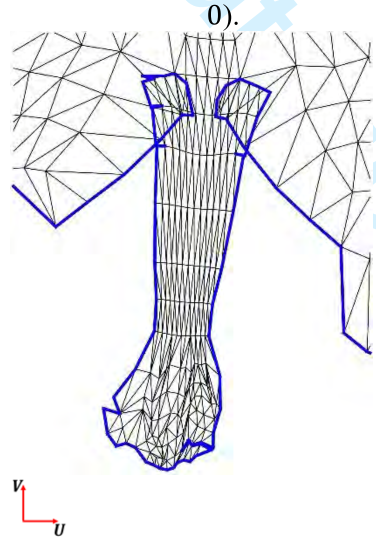


(b) *Cow* parameterization ($\alpha = 0.01$).



(c) *Cow* parameterization ($\alpha = 0$).





Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

(d) *Cow* parameterization ($\alpha =$ 0). Zoom into head.

(e) *Cow* parameterization ($\alpha =$ 0). Zoom into tail.

Figure 8: Parameterization results for the *Cow* dataset.

4.3. *Results for other datasets*

Fig. 11 presents the parameterization results of the proposed algorithm for the *Sliced-Glove*, *Fandisk*, *Foot* and *Bull* datasets. With an $\alpha = 0.5$, the algorithm converged in all the presented cases to valid parameterizations (except for the *Bull* parameterization which is not bijective). Table 3 presents a quantification of the performance of our algorithm for the case studies. A random initial parameterization U^0_{Rand} is used and in all the test cases, $\|\nabla F^*\| < 10^{-8}$ and the lowest eigenvalue of the Hessian is nonnegative indicating that a local minimum has been reached and the algorithm has converged. All the resulting parameterizations are locally bijective (no triangle flips).

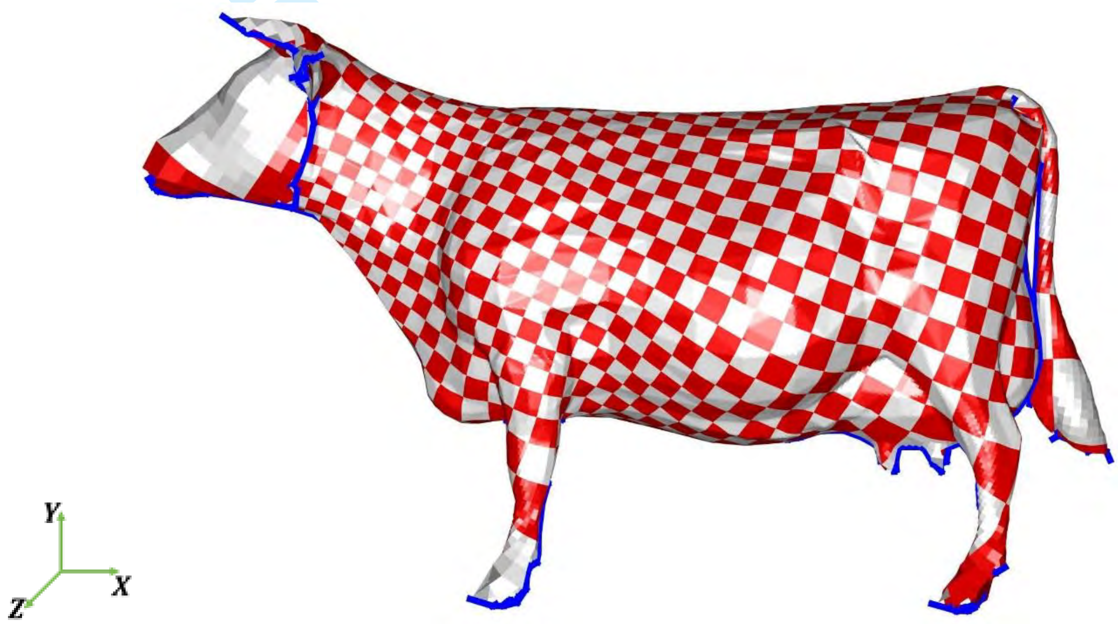
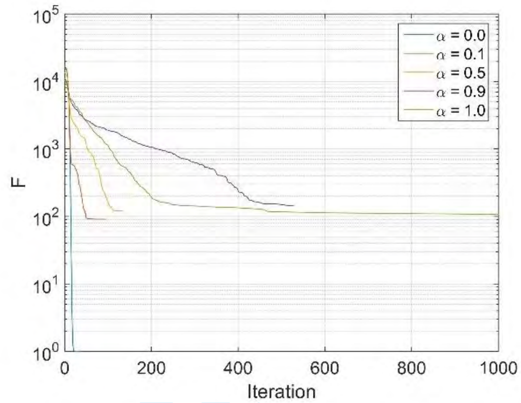
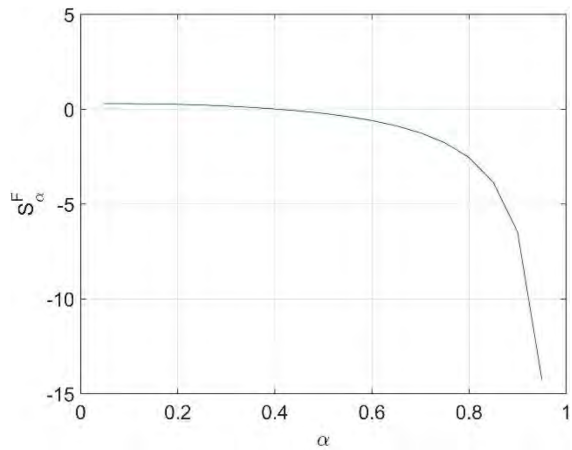


Figure 9: Texture map for the *Cow* dataset ($\alpha = 0$). Area distortion is more noticeable in the head, legs and tail.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072



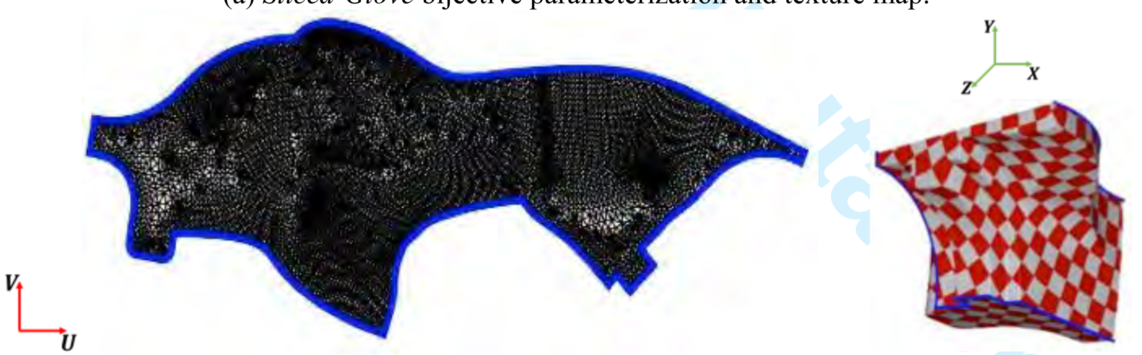
(a) Evolution of F for different α values.



(b) Relative sensitivity S^F_α.

Figure 10: *Cow* dataset. Sensitivity analysis of F with respect to α. For area-preserving parameterizations, (a) the algorithm evidences slower convergence for higher values of α and (b) the penalty function F is highly sensitive in the area preserving ($\alpha \to 1$) side.

(a) *Sliced-Glove* bijective parameterization and texture map.

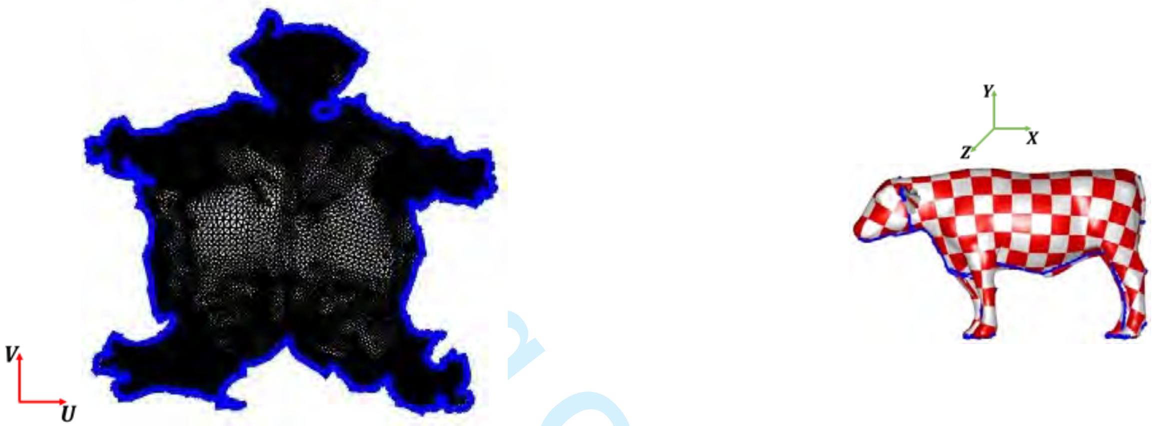


(b) *Fandisk* bijective parameterization and texture map.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

(c) *Foot* bijective parameterization and texture map.



(d) *Bull* non-bijective parameterization and texture map.

Figure 11: Parameterization results for several datasets and their corresponding texture map. All meshes are 2-manifolds with border.

Dataset	$\boldsymbol{n}$	$\boldsymbol{\alpha}$	$\|\|\boldsymbol{\nabla F}\|\|$	**Result**
Beetle	988	0.1	2×10^{-12}	Fig. 5
Cow	3195	0	4×10^{-11}	Fig. 9
Sliced-Glove	985	0.5	7×10^{-11}	Fig. 11(a)
Fandisk	6699	0.5	4×10^{-11}	Fig. 11(b)
Foot	10211	0.5	6×10^{-11}	Fig. 11(c)
Bull	17918	0.1	4×10^{-9}	Fig. 11(d)

Table 3: Quantification of our algorithm performance for the cases studied (n: number of iterations, α: weight of area - preserving criterion, $||\boldsymbol{\nabla F}||$: gradient of objective function).

4.4. Comparison with competitor algorithms

To compare our algorithm with other Mesh Parameterization algorithms, the GraphiteLE software (https://gforge.inria.fr/frs/?group_id=1465, accessed 01 August 2016) is used to run the ABF and the LSCM algorithms, while the MATLAB version of ARAP (see Table 2) status of 01 August 2016, is used for computing such parameterization. The algorithms are run with default parameters and no modifications to the routines have been made. Furthermore, ARAP is run with a valid initial parameterization (ABF by default) while our algorithm is run with a random initial guess.

Engineering Computations

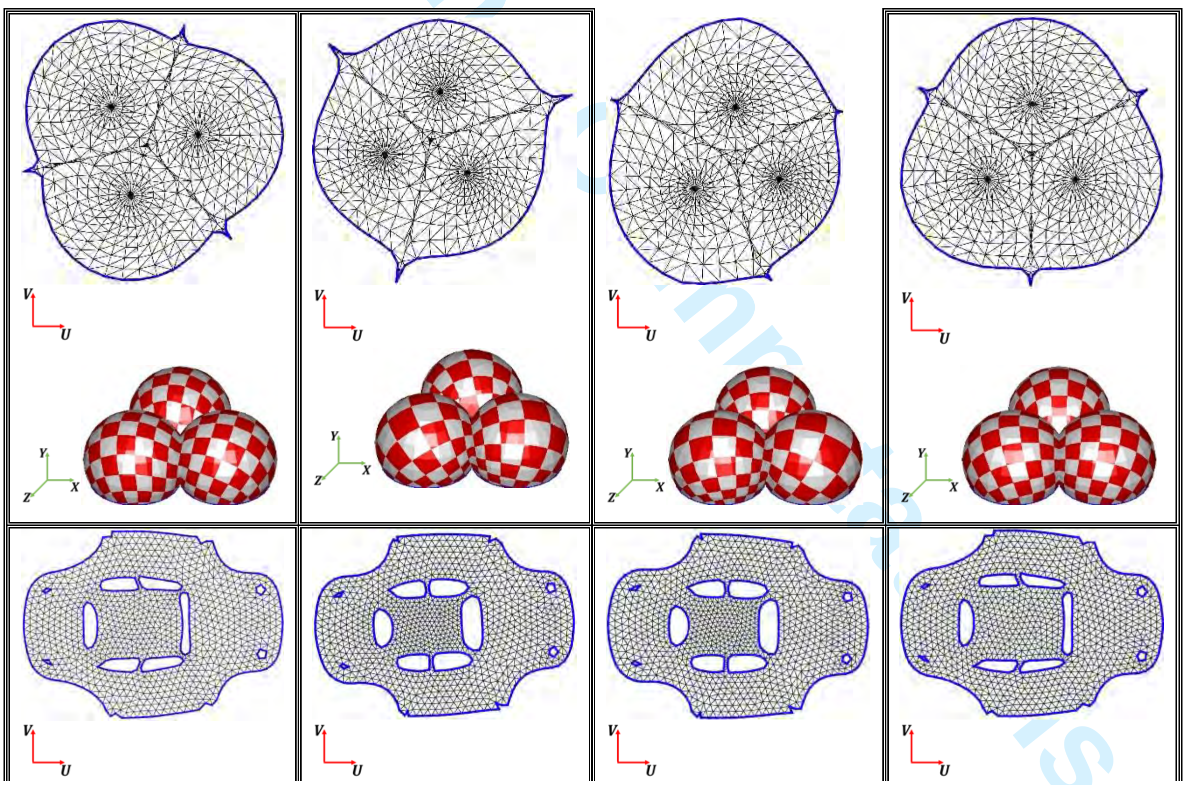
Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Fig. 12 presents the parameterization results obtained by our algorithm vs. ABF, LSCM and ARAP for the *Balls*, *Beetle*, *Cow* and *Gargoyle* datasets. We do not measure execution times because: (1) the algorithms are written in different programming languages (C and MATLAB), (2) ARAP has its own data which requires a pre-processing, (3) in general, we cannot provide an even field for algorithm comparison. Table 4 and Figure 12 present the results for the different algorithms. For the four datasets, ABF and our algorithm present no triangle flips, while LSCM and ARAP invert some triangles through the process. Our algorithm results in globally bijective parameterizations for the *Balls*, *Beetle* and *Gargoyle* datasets. None of the algorithms reaches a globally bijective parameterization for the *Cow* dataset.

	Our Algorithm		**ABF**		**LSCM**		**ARAP**	
Dataset	Triangle Flips	Global Overlaps	Triangle Flips	Global Overlaps	Triangle Flips	Global Overlaps	Triangle Flips	Global Overlaps
Balls	No	No	No	No	Yes	Yes	No	No
Beetle	No	No	No	No	No	No	No	No
Cow	No	Yes	No	Yes	Yes	Yes	Yes	Yes
Gargoyle	No	No	No	No	No	No	Yes	Yes

Table 4: Appraisal of the parameterization results for our algorithm, ABF, LSCM and ARAP. A necessary condition for a valid parameterization is the absence of triangle flips and global overlaps.



Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE), Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

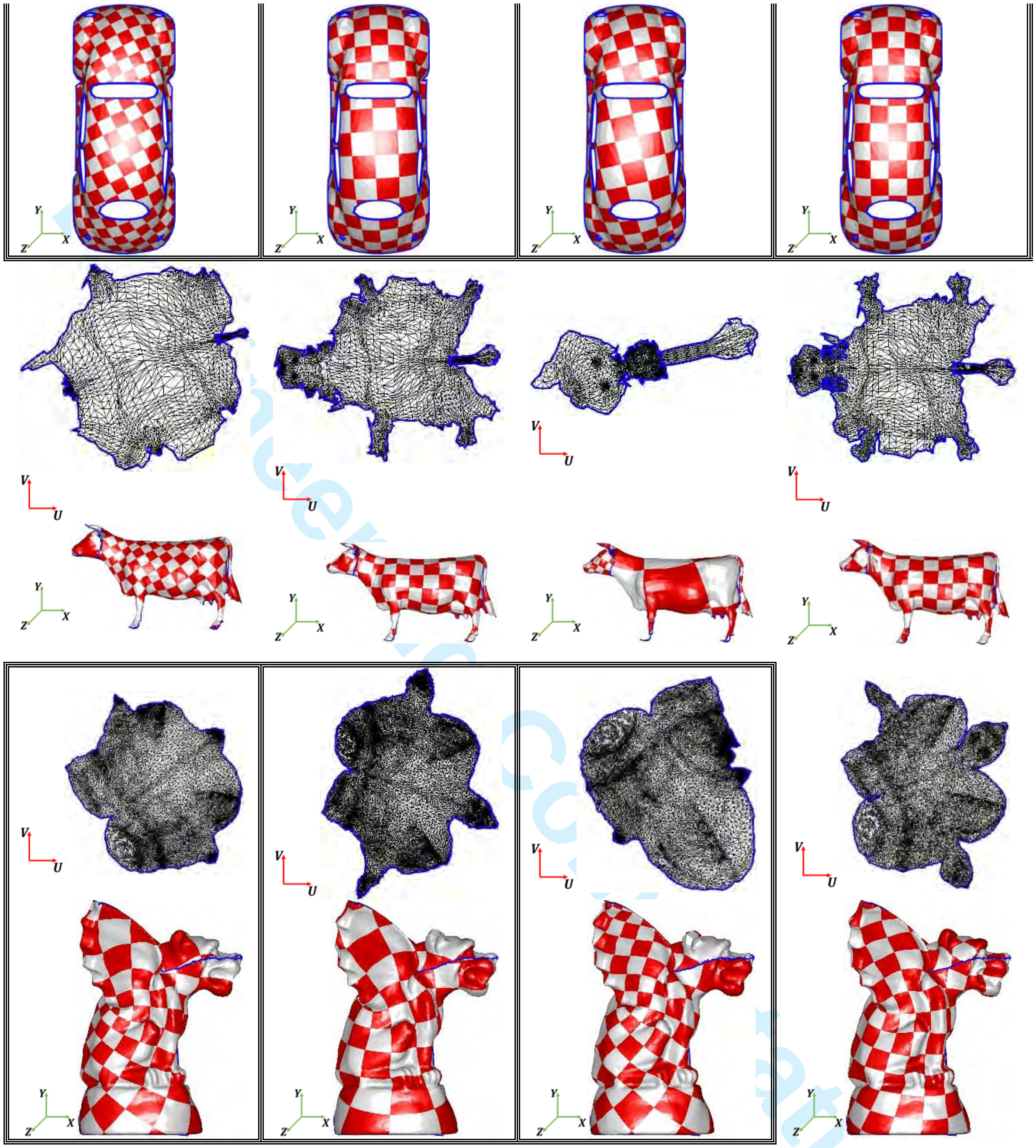


Figure 12: Parameterization results for the *Balls*, *Beetle*, *Cow* and the *Gargoyle* datasets using our algorithm and several state of the art Mesh Parameterization algorithms. Legal parameterizations are enclosed with a framed cell.

4.5. Complexity analysis

The time complexity of our algorithm is discussed in this section. In Ref. (Ueda and Yamashita, 2012), a complexity analysis of the LM algorithm has been presented which shows that LM iterates $\mathcal{O}(\ln \varepsilon^{-1})$ times. In our algorithm, for a mesh with m triangles and n nodes each LM iteration must perform the following operations:

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

1) Compute $F(U^k)$, $\nabla F(U^k)$ and $\mathcal{H}[F(U^k)]$.
2) Solve the linear system $\left(\mathcal{H}[F(U^k)] + \lambda^k I_{2n}\right)^{-1} \nabla F(U^k)$ as per Eq. (9).

In the first step of the LM iteration, $F(U^k)$, $\nabla F(U^k)$ and $\mathcal{H}[F(U^k)]$ are computed by adding the distortion D^i_{area} and D^i_{angle} and their corresponding derivatives for each individual triangle. The cost of this operation is $\mathcal{O}(m)$. For the linear system in the second step, the matrix $\mathcal{H}[F(U^k)] + \lambda^k I_{2n}$ is a $2n \times 2n$ symmetric sparse matrix whose nonzero elements correspond to adjacent nodes in the mesh as discussed in section 3.5. The solution of this linear system costs $\mathcal{O}(nz)$ (where nz is the number of nonzeros in the matrix $\mathcal{H}[F(U^k)] + \lambda^k I_{2n}$). Due to the Euler characteristic of triangular meshes, $nz \approx 28n$. Hence, the complexity order for the linear system solution becomes $\mathcal{O}(n)$ (Botsch *et al.*, 2005).

In summary, the order of our algorithm is of the same order of the LM algorithm times the internal loop i.e., $\mathcal{O}(\ln \varepsilon^{-1})\left(\mathcal{O}(m) + \mathcal{O}(n)\right)$. However, for a fixed ε and assuming (by the Euler characteristic) that $\mathcal{O}(m) = \mathcal{O}(n)$, our algorithm becomes of the order $\mathcal{O}(c \cdot n)$ (where c is the number of iterations of the LM algorithm before convergence). Table 3 presents a comparison of computational complexities for several Mesh Parameterization algorithms.

Reference	Algorithm	Complexity
N/A	Our LM Mesh Parameterization	$\mathcal{O}(c \cdot n)$
(Donoho and Grimes, 2003)	Classic HLLE	$\mathcal{O}(c \cdot n)$
(Floater, 1997)	Floater Parameterization	$\mathcal{O}(c \cdot n)$
(Yoshizawa *et al.*, 2004)	Stretch Minimizing Parameterization	$\mathcal{O}(c_1 \cdot c_2 \cdot n)$
(Desbrun *et al.*, 2002)	Intrinsic Parameterization	$\mathcal{O}(c \cdot n)$
(Lévy *et al.*, 2002)	LSCM	$\mathcal{O}(c \cdot n)$
(Lee *et al.*, 2002)	Virtual Boundary Parameterization	$\mathcal{O}(c_1 \cdot n + c_2 \cdot \lvert\partial M\rvert)$
(Zayer *et al.*, 2007)	Linear ABF	$\mathcal{O}(c \cdot n)$
(Liu *et al.*, 2008)	ASAP	$\mathcal{O}(c \cdot n)$
(Liu *et al.*, 2008)	ARAP	$\mathcal{O}(c_1 \cdot c_2 \cdot n + c_3 \cdot \lvert\partial M\rvert)$
(Sheffer and de Sturler, 2001)	ABF	$\mathcal{O}(c_1 \cdot c_2 \cdot n)$
(Kharevych *et al.*, 2006)	Discrete Conformal Mappings	$\mathcal{O}(c_1 \cdot c_2 \cdot n)$
(Smith and Schaefer, 2015)	Free Boundary Parameterization	$\mathcal{O}\left(c(n + \lvert\partial M\rvert)\right)$

Table 3: Computational time complexity of several Mesh Parameterization algorithms. c denotes the number of iterations before convergence for that particular algorithm while $|\partial M|$ denotes the number of vertices at the boundary of M.

5. Conclusions

This article presents an algorithm for parameterizing a triangular mesh M of a 2-manifold with non-empty border embedded in $\mathbb{R}^3$. The proposed algorithm consists of mapping each triangle individually to the plane Z=0 by a rigid transformation η and then mapping it to the global parameterization ϕ by an affine mapping ψ. The parameterization $U = \phi(X)$ is obtained by minimizing the weighted area and angle distortion of ψ (which also penalizes triangle flips) with the LM algorithm. The complexity analysis of our algorithm shows asymptotic linear behavior $\mathcal{O}(c \cdot n)$ in the number of vertices which makes our method comparable to most mesh parameterization that are also asymptotically linear in time as illustrated in table 3. Our algorithm presents the advantage

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

over other nonlinear-gradient parameterization algorithms of not requiring an initial valid parameterization.

A weighting parameter α is introduced in the penalty distortion function which allows tuning by the user to favour area against angle preservation turning a non-bijective parameterization into a bijective one in specific cases. Our sensitivity analysis shows that our penalty function is very sensitive in the domain of area preservation ($\alpha \rightarrow 1$). Our experiments show that global overlaps are more frequent when preserving areas than when preserving angles, encouraging angle preservation. It must be remarked that a sensitivity analysis assesses the influence of the parameters on the penalty function $\boldsymbol{F}$ and not the goodness of $\boldsymbol{F}$ for the problem at hand.

Compared to other Mesh Parameterization algorithms in general, our algorithm converged presenting correct results across the datasets, rendering low distortion, non-overlapping and valid parameterizations except for highly non-developable datasets (i.e., *Cow* and *Bull*).

4.1. *Ongoing work*

Segmentation of large meshes into smaller ones increases the probability of finding bijective individual parameterizations for the smaller ones. Therefore, it is of interest to explore mesh segmentation as a necessary step for mesh parameterization.

Bijectivity of the resulting parameterization relates to: (1) local overlaps (triangle flips) and (2) global overlaps. Although not proven, all of our experiments present no triangle flips, partially addressing the problem of local overlaps. However, global bijectivity is a non-trivial constraint which increases the computational complexity of the algorithm as shown in Ref. (Smith and Schaefer, 2015). Therefore, further work is required on this aspect.

References

ALICE project-team (2008), “Unwrapped meshes”, available at: http://alice.loria.fr/index.php/software/7-data/37-unwrapped-meshes.html (accessed 10 February 2016).

Athanasiadis, T., Zioupos, G. and Fudos, I. (2013), “Efficient computation of constrained parameterizations on parallel platforms”, *Computers & Graphics*, Vol. 37 No. 6, pp. 596-607.

Belkin, M. and Niyogi, P. (2003), “Laplacian eigenmaps for dimensionality reduction and data representation”, *Neural Computation*, Vol. 15 No. 6, pp. 1373-1396.

Ben-Chen, M., Gotsman, C. and Bunin, G. (2008), “Conformal flattening by curvature prescription and metric scaling”, *Computer Graphics Forum*, Vol. 27 No. 2, pp. 449-458.

Botsch, M., Bommes, D. and Kobbelt, L. (2005), “Efficient linear system solvers for mesh processing”, in Martin, R., Bez, H. and Sabin, M. (Ed), *Mathematics of Surfaces XI: 11th IMA International Conference, Loughborough, UK, September 5-7, 2005. Proceedings*, Springer Berlin Heidelberg, Berlin, pp. 62-83.

Chen, Z., Liu, L., Zhang, Z. and Wang, G. (2007), “Surface parameterization via aligning optimal local flattening”, in *Proceedings of the 2007 ACM Symposium on Solid and Physical Modeling (SPM '07)*, ACM, Beijing, pp. 291-296.

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Degener, P., Meseth, J. and Klein, R. (2003), "An adaptable surface parameterization method", in *Proceedings of the 12th International Meshing Roundtable*, pp. 201-213.

Desbrun, M., Meyer, M. and Alliez, P. (2002), "Intrinsic parameterizations of surface meshes", *Computer Graphics Forum*, Vol. 21. No. 3, pp. 209-218.

Donoho, D.L. and Grimes, C. (2003), "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data", *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 100 No. 10, pp. 5591-5596.

Edgar, T.F. and Himmelblau, D.M. (2001), *Optimization of Chemical Processes, Second Edition*, MacGraw-Hill, New York, NY.

Floater, M.S. (1997), "Parameterization and smooth approximation of surface triangulations", *Computer Aided Geometric Design*, Vol. 14 No. 3, pp. 231-250.

Hormann, K. and Greiner, G. (2000), "MIPS: An efficient global parametrization method", in Laurent, P.J., Sablonniere, P. and Schumaker, L.L. (Ed), *Curve and Surface Design: Saint-Malo 1999*, Vanderbilt University Press, Nashville, pp. 153-162.

Hormann, K., Lévy, B. and Sheffer, A. (2007), "Mesh Parameterization: Theory and Practice", in *SIGGRAPH 2007 Course Notes*, ACM Press, San Diego, CA, pp. vi+115.

Kharevych, L., Springborn, B. and Schröder, P. (2006), "Discrete conformal mappings via circle patterns", *ACM Transactions on Graphics*, Vol. 25 No. 2, pp. 412-438.

Lévy, B., Petitjean, S., Ray, N. and Maillot, J. (2002), "Least squares conformal maps for automatic texture atlas generation", *ACM Transactions on Graphics*, Vol. 21 No 3, pp. 362-371.

Lee, Y., Kim, H.S. and Lee, S. (2002), "Mesh parameterization with a virtual boundary", *Computers & Graphics*, Vol. 26, pp. 677-686.

Li, B., Zhang, X., Zhou, P. and Hu, P. (2010), "Mesh parameterization based on one-step inverse forming", *Computer-Aided Design*, Vol. 42 No. 7, pp. 633-640.

Li, Z., Jin, Y., Jin, X. and Ma, L. (2012), "Approximate straightest path computation and its application in parameterization", *The Visual Computer*, Vol. 28 No. 1, pp. 63-74.

Liu, L., Zhang, L., Xu, Y., Gotsman, C. and Gortler, S.J. (2008), "A local/global approach to mesh parameterization", in *Proceedings of the Symposium on Geometry Processing*, Eurographics Association, Aire-la-Ville, pp. 1495-1504.

Mejia, D., Ruiz-Salguero, O. and Cadavid, C.A. (2016), "Hessian eigenfunctions for triangular mesh parameterization", in *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pp. 75-82.

Myles, A. and Zorin, D. (2012), "Global parametrization by incremental flattening", *ACM Transactions on Graphics*, Vol. 31 No. 4, pp. 109:1-109:11.

Myles, A. and Zorin, D. (2013), "Controlled-distortion constrained global parametrization", *ACM Transactions on Graphics*, Vol. 32 No. 4, pp. 105:1-105:14.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Nieser, M., Palacios, J., Polthier, K. and Zhang, E. (2012), "Hexagonal global parameterization of arbitrary surfaces", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, No. 6, pp. 865-878.

Papadimitriou, C.H. and Steiglitz, K. (1982), *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Inc., Upper Saddle River, NJ.

Pietroni, N., Massimiliano, C., Cignoni, P. and Scopigno, R. (2011), "An interactive local flattening operator to support digital investigations on artwork surfaces", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17 No. 12, pp. 1989-1996.

Ravindran, A., Ragsdell, K.M. and Reklaitis, G.V. (2007), "Functions of several variables", in *Engineering Optimization: Methods and Applications*, John Wiley & Sons, Inc, Hoboken, NJ, pp. 78-148.

Roweis, S.T. and Saul, L.K. (2000), "Nonlinear dimensionality reduction by locally linear embedding", *Science*, Vol. 290, pp. 2323-2326.

Ruiz, O.E., Mejia, D., Cadavid, C.A. (2015), "Triangular mesh parameterization with trimmed surfaces", *International Journal on Interactive Design and Manufacturing*, Vol. 9 No. 4, pp. 303-316.

Sander, P.V., Gortler, S.J., Snyder, J. and Hoppe, H. (2002), "Signal-specialized parametrization", in *Proceedings of the 13th Eurographics Workshop on Rendering*, Eurographics Association, Aire-la-Ville, pp. 87-98.

Sheffer, A. and de Sturler, E. (2001), "Parameterization of faceted surfaces for meshing using angle-based flattening", *Engineering with Computers*, Vol. 17 No. 3, pp. 326-337.

Sheffer, A. and Hart, J.C. (2002), "Seamster: Inconspicuous low-distortion texture seam layout", in *Proceedings of the Conference on Visualization '02*, IEEE Computer Society, Washington, DC, pp. 291-298.

Sheffer, A. Lévy, B., Mogilnitsky, M. and Bogomyakov, A. (2005), "ABF++: Fast and robust angle based flattening", *ACM Transactions on Graphics*, Vol. 24 No. 2, pp. 311-330.

Sheffer, A. Praun, E. and Rose, K. (2006), "Mesh parameterization methods and their applications", Foundations and Trends® in Computer Graphics and Vision, Vol. 2 No. 2, pp. 105-171.

Smith, J. and Schaefer, S. (2015), "Bijective parameterization with free boundaries", *ACM Transactions on Graphics*, Vol. 34 No. 4, pp. 70:1-70:9.

Springborn, B., Schröder, P. and Pinkall, U. (2008), "Conformal equivalence of triangle meshes", *ACM Transactions on Graphics*, Vol. 27 No. 3, pp. 77:1-77:11.

Su, K., Cui, L., Qian, K., Lei, N., Zhang, J., Zhang, M. and Gu, X.D. (2016), "Area-preserving mesh parameterization for poly-annulus surfaces based on optimal mass transportation", *Computer Aided Geometric Design*, in Press, Corrected Proof.1

Sun, X. and Hancock, E.R. (2008), "Quasi-isometric parameterization for texture mapping", *Pattern Recognition*, Vol. 41 No. 5, pp. 1732-1743.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072

Tenenbaum, J.B., de Silva, V. and Langford, J.C. (2000), "A global geometric framework for nonlinear dimensionality reduction", *Science*, Vol. 290 No. 5500, pp. 2319-2323.

Ueda, K. and Yamashita, N. (2012), "Global complexity bound analysis of the Levenberg-Marquardt method for nonsmooth equations and its application to the nonlinear complementarity problem", *Journal of Optimization Theory and Applications*, Vol. 152 No. 2, pp. 450-467.

Wang, Z., Luo, Z.-X., Zhang, J.-L. and Saucan, E. (2016), "ARAP++: an extension of the local/global approach to mesh parameterization", *Frontiers of Information Technology & Electronic Engineering*, Vol. 17 No. 6, pp. 501-515.

Yoshizawa, S., Belyaev, A. and Seidel, H.-P. (2004), "A fast and simple stretch-minimizing mesh parameterization", in *Proceedings of the Shape Modeling International 2004*, IEEE Computer Society, Washington, DC, pp. 200-208.

Zayer, R., Lévy, B. and Seidel, H.-P. (2007), "Linear angle based parameterization", in *Proceedings of the Fifth Eurographics Symposium on Geometry Processing*, Eurographics Association, Aire-la-Ville, pp. 135-141.

Zayer, R., Rössl, C. and Seidel, H.-P. (2005), "Setting the boundary free: A composite approach to surface parameterization", in *Proceedings of the Third Eurographics Symposium on Geometry Processing*, Eurographics Association, Aire-la-Ville.

Zhang, Z. and Zha, H. (2002), "Principal manifolds and nonlinear dimension reduction via local tangent space alignment", *SIAM Journal of Scientific Computing*, Vol. 26, pp. 313-338.

Zhao, X., Su, Z., Gu, X.D., Kaufman, A., Sun, J., Gao, J. and Luo, F. (2013), "Area-preservation mapping using optimal mass transport", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 19 No. 12, pp. 2838-2847.

Zhu, X.-F., Hu, P., Ma, Z.-D., Zhang, X., Li, W., Bao, J. and Liu, M. (2013), "A new surface parameterization method based on one-step inverse forming for isogeometric analysis-suited geometry", *The International Journal of Advanced Manufacturing Technology*, Vol. 65 No. 9, pp. 1215-1227.

Zou, G., Hu, J., Gu, X. and Hua, J. (2011), "Authalic parameterization of general surfaces using lie advection", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17 No. 12, pp. 2005-2014.

Engineering Computations

Daniel Mejia (Lab. CAD CAM CAE, Diego A. Acosta (Processes Group) Oscar Ruiz-Salguero. (Lab. CAD CAM CAE). Univ EAFIT, Colombia

Weighted area / angle distortion minimization for Mesh Parameterization. Daniel Mejia, Diego A. Acosta, Oscar Ruiz-Salguero, J. Eng. Computations, Emerald Isight, http://www.emeraldinsight.com/doi/full/10.1108/EC-02-2016-0072 , ISSN: 0264-4401, 2017, v34, n6, doi: 10.1108/EC-02-2016-0072