On Planar Shape Interpolation with Logarithmic Metric Blending Supplementary Material

ALON FELDMAN, Technion - Israel Institute of Technology, Israel
MIRELA BEN-CHEN, Technion - Israel Institute of Technology, Israel

CCS Concepts: • **Computing methodologies** → **Shape analysis**.

Additional Key Words and Phrases: geometry processing, shape interpolation

ACM Reference Format:
Alon Feldman and Mirela Ben-Chen. 2025. On Planar Shape Interpolation with Logarithmic Metric Blending Supplementary Material. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '25), August 10–14, 2025, Vancouver, BC, Canada.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3721238.3730697

1 Linear Metric Interpolation Distortion

Proposition 1.1. *Given $g_t = (1-t)I + t \cdot g_\phi$, the eigenvalues of the intermediate metric g_t are linear interpolations of the eigenvalues of the target metric g_ϕ, and are linear functions of t.*

Proof. We express g_ϕ in the diagonal form $V\Lambda_\phi V^\top$, where V is the orthonormal eigenvector matrix and Λ_ϕ is the diagonal matrix of eigenvalues. We can also diagonalize I as $VIV^\top$, which is valid because V satisfies $VV^\top = I$.

$$\begin{aligned} g_t &= (1-t)VIV^\top + tV\Lambda_\phi V^\top \\ &= V\big((1-t)I + t\Lambda_\phi\big)V^\top. \end{aligned} \tag{1}$$

From this, we find that the eigenvalues of g_t are linearly interpolated as:

$$\Lambda_t = (1-t)I + t\Lambda_\phi,$$

and the i-th eigenvalue $\lambda_{t,i}$, is linear function of t:

$$\lambda_{t,i} = (1-t) + t\lambda_{\phi,i} = 1 + t(\lambda_{\phi,i} - 1). \tag{2}$$

□

Corollary 1.2. *Given $g_t = (1-t)I + t \cdot g_\phi$, the conformal and area distortions induced by g_t are:*

$$K_t(p) = \sqrt{\frac{\lambda_{t,1}}{\lambda_{t,2}}} = \sqrt{\frac{1 + t(\lambda_{\phi,1} - 1)}{1 + t(\lambda_{\phi,2} - 1)}}, \tag{3}$$

$$D_t(p) = \sqrt{\lambda_{t,1} \cdot \lambda_{t,2}} = \sqrt{\big(1 + t(\lambda_{\phi,1} - 1)\big)\big(1 + t(\lambda_{\phi,2} - 1)\big)}$$

Authors' Contact Information: Alon Feldman, Technion - Israel Institute of Technology, Haifa, Israel, alon.feldman99@gmail.com; Mirela Ben-Chen, Technion - Israel Institute of Technology, Haifa, Israel, mirela@cs.technion.ac.il.

SIGGRAPH Conference Papers '25, August 10–14, 2025, Vancouver, BC, Canada

ACM ISBN 979-8-4007-1540-2/2025/08
https://doi.org/10.1145/3721238.3730697

2 Square-Root Metric Interpolation Distortion

Proposition 2.1. *Given $g_t = ((1-t)\cdot I^{0.5} + t\cdot g_\phi^{0.5})^2$, the eigenvalues of the intermediate metric g_t can be expressed as:*

$$\lambda_{t,i} = (1 + t(\sqrt{\lambda_{\phi,i}} - 1))^2.$$

Proof. We express g_ϕ in the diagonal form $V\Lambda_\phi V^\top$, where V is the orthonormal eigenvector matrix and Λ_ϕ is the diagonal matrix of eigenvalues. We can also diagonalize I as $VIV^\top$, which is valid because V satisfies $VV^\top = I$.

$$\begin{aligned} g_t &= \Big((1-t)\cdot (VIV^\top)^{0.5} + t\cdot (V\Lambda_\phi V^\top)^{0.5}\Big)^2 \\ &= \Big((1-t)\cdot (VI^{0.5}V^\top) + t\cdot (V\Lambda_\phi^{0.5}V^\top)\Big)^2 \\ &= \Big(V((1-t)I^{0.5} + t\Lambda_\phi^{0.5})V^\top\Big)^2 \\ &= V\Big((1-t)I^{0.5} + t\Lambda_\phi^{0.5}\Big)^2 V^\top. \end{aligned} \tag{4}$$

Hence, the eigenvalues of g_t are:

$$\Lambda_t = \Big((1-t)I^{0.5} + t\Lambda_\phi^{0.5}\Big)^2,$$

and the i-th eigenvalue $\lambda_{t,i}$, is a quadratic function of t:

$$\lambda_{t,i} = ((1-t) + t\sqrt{\lambda_{\phi,i}})^2 = (1 + t(\sqrt{\lambda_{\phi,i}} - 1))^2 \tag{5}$$

□

Corollary 2.2. *Given $g_t = ((1-t)\cdot I^{0.5} + t\cdot g_\phi^{0.5})^2$, the conformal and area distortions induced by g_t are:*

$$K_t(p) = \sqrt{\frac{\lambda_{t,1}}{\lambda_{t,2}}} = \frac{1 + t(\sqrt{\lambda_{\phi,1}} - 1)}{1 + t(\sqrt{\lambda_{\phi,2}} - 1)}, \tag{6}$$

$$D_t(p) = \sqrt{\lambda_{t,1}\cdot\lambda_{t,2}} = (1 + t(\sqrt{\lambda_{\phi,1}} - 1))(1 + t(\sqrt{\lambda_{\phi,2}} - 1)).$$

3 Lagrange, Smoothness, and Symmetry Properties

Lagrange Property. This property follows directly from the definition of the interpolated metrics g_t. At $t = 0$, all blending schemes — linear, square-root, and logarithmic — reduce to the identity metric $g_0 = I$, corresponding to the undeformed source shape. Similarly, at $t = 1$, the interpolated metric becomes the pullback metric $g_1 = g$, associated with the deformation map ϕ from source to target. Since the interpolation recovers the exact input metrics at both endpoints, we assume realizations that faithfully reproduce the corresponding embeddings.

Smoothness. Let $\phi : S_0 \to S_1$ be a continuously differentiable map. Then the pullback metric g_ϕ is differentiable over the domain S_0. Consequently, the interpolated metric g_t, constructed via any of the blending schemes (linear, square-root, or logarithmic), depends smoothly on the interpolation parameter t, and its derivative $\frac{dg_t}{dt}$ exists. The smoothness of the realization stage further depends on the specific parameterization method used. For instance, Chen et al. [2013] demonstrated that conformal parameterization using CETM leads to a smooth evolution with respect to t. Similarly, embedding using LSCM will lead to a smooth result. Note that in both cases, the degrees of freedom of rigid motion (rotation and translation) need to be specified smoothly for this to hold. For example, in LSCM the two fixed vertices should remain the same. As ARAP can theoretically converge to a local minima, the result may not always be smooth. In practice, we have not encountered this problem.

Symmetry. To establish symmetry, it suffices to show that the interpolated metrics are equal when represented in the same domain, specifically:

$$g_{\phi(t)} = g_{\phi^{-1}(1-t)},$$

where both sides are expressed with respect to the domain S_0.

Let $\phi : S_0 \to S_1$ be a differentiable map with Jacobian $J_\phi = U\Sigma V^T$, so that the pullback metric is

$$g_\phi = J_\phi^\top J_\phi = V\Sigma^2 V^T.$$

Let $\psi = \phi^{-1} : S_1 \to S_0$, with Jacobian $J_\psi = V\Sigma^{-1}U^T$, yielding the pullback metric

$$g_\psi = J_\psi^\top J_\psi = U\Sigma^{-2}U^T.$$

We define time-dependent maps $\phi(t) : S_0 \to S_t$ and $\psi(\tilde{t}) : S_1 \to \tilde{S}_t$, with $\tilde{t} = 1 - t$, and corresponding interpolated metrics:

$$g_{\phi(t)} = V\Lambda_{\phi(t)}V^T, \quad g_{\psi(\tilde{t})} = U\Lambda_{\psi(\tilde{t})}U^T,$$

where:

$$\Lambda_{\phi(t)} = \text{blend}_t(I, \Sigma^2), \quad \Lambda_{\psi(\tilde{t})} = \text{blend}_{\tilde{t}}(I, \Sigma^{-2}).$$

To express $g_{\psi(\tilde{t})}$ with respect to S_0, we pull it back via ϕ:

$$g^{S_0}_{\phi^{-1}(\tilde{t})} = J_\phi^\top U\Lambda_{\psi(\tilde{t})}U^T J_\phi = V\Sigma\Lambda_{\psi(\tilde{t})}\Sigma V^T.$$

Thus, proving symmetry reduces to verifying:

$$\Sigma\Lambda_{\psi(1-t)}\Sigma = \Lambda_{\phi(t)}.$$

Linear Blending.

$$\begin{aligned}\Sigma \cdot \Lambda_{\psi(1-t)} \cdot \Sigma &= \Sigma \cdot \left[(1-(1-t))I + (1-t)\Sigma^{-2}\right] \cdot \Sigma \\ &= \Sigma \cdot \left[tI + (1-t)\Sigma^{-2}\right] \cdot \Sigma \\ &= t\Sigma^2 + (1-t)I = \Lambda_{\phi(t)}.\end{aligned} \tag{7}$$

Logarithmic Blending.

$$\begin{aligned}\Sigma \cdot \Lambda_{\psi(1-t)} \cdot \Sigma &= \Sigma \cdot \Sigma^{-2(1-t)} \cdot \Sigma \\ &= \Sigma^{2t} = \Lambda_{\phi(t)}.\end{aligned} \tag{8}$$

Square-Root Blending.

$$\begin{aligned}\Sigma \cdot \Lambda_{\psi(1-t)} \cdot \Sigma &= \Sigma \cdot \left[tI + (1-t)\Sigma^{-1}\right]^2 \cdot \Sigma \\ &= \left[t\Sigma + (1-t)I\right]^2 = \Lambda_{\phi(t)}.\end{aligned} \tag{9}$$

4 Log Edge Length Interpolation From Discrete Conformal Factors

4.1 Discrete Conformal Factors

A pair of triangular meshes $(S, \tilde{S})$ are considered Conformally Equivalent (CETM) [Springborn et al. 2008] if they have the same connectivity and if there exists a set of discrete conformal factors $u : V \to \mathbb{R}$ such that their edge lengths $(l, \tilde{l})$ satisfy the relation:

$$\tilde{l}_{ij} = e^{\frac{u_i+u_j}{2}} l_{ij}.$$

This is a discrete analog of Conformally Equivalent Domains. Discrete conformal factors are fundamental in establishing a combinatorial notion of conformal structures, as explored in Luo's combinatorial Yamabe flow [Luo 2004].

A pair of triangles $f_{ijk} \in S, \tilde{f}_{ijk} \in \tilde{S}$ are always considered CETM (as long as their orientation is preserved), as appropriate discrete conformal factors $u_f : V_{\{i,j,k\}} \to \mathbb{R}$ can be extracted using the logarithm of the edge length ratios [Springborn et al. 2008]:

$$u_{f,i} = \ln\left(\frac{\tilde{l}_{ij}}{l_{ij}} \cdot \frac{\tilde{l}_{ki}}{l_{ki}} \cdot \frac{l_{jk}}{\tilde{l}_{jk}}\right).$$

Given a pair of meshes with the same connectivity, the discrete conformal factors u_f exist per face $(f_{ijk}, \tilde{f}_{ijk})$. However, a global $u : V \to \mathbb{R}$ exists if and only if the pair of meshes are CETM.

4.2 Linear Blending of Discrete Conformal Factors

With the notion of discrete conformal factors and CETM, we can leverage the equivalence between the geometrical blending of metrics and the linear blending of conformal factors, as established in our paper Proposition 3.3, Equation (10).

Since each pair of faces is CETM, we propose linearly blending the corresponding discrete conformal factors:

$$u(f, t) = t \cdot u_f$$

. This induces new intermediate edge lengths:

$$l_{ij}(f, t) = e^{\frac{t\cdot u_{f,i}+t\cdot u_{f,j}}{2}} l_{ij}. \tag{10}$$

We show that the intermediate edge lengths are consistent between adjacent faces as the exponential factor can be expressed by the edge lengths of source and target, eliminating u_f from the expression.

$$\tilde{l}_{ij} = e^{\frac{u_{f,i}+u_{f,j}}{2}} l_{ij}$$

$$\frac{\tilde{l}_{ij}}{l_{ij}} = e^{\frac{u_{f,i}+u_{f,j}}{2}}$$

Rewriting expression (10) for the intermediate edge lengths:

$$l_{ij}(f, t) = e^{\frac{t\cdot u_{f,i}+t\cdot u_{f,j}}{2}} l_{ij} = \left(e^{\frac{u_{f,i}+u_{f,j}}{2}}\right)^t l_{ij} = \left(\frac{\tilde{l}_{ij}}{l_{ij}}\right)^t l_{ij}.$$

This shows that the desired intermediate edge lengths are provided by the simple geometric blending of the source and target edge lengths:

$$l_{ij}(t) = l_{ij}^{1-t}\ \tilde{l}_{ij}^{t}. \tag{11}$$

5 Local/Global As-Equiareal-As-Possible Singular Values and Energy

Based on the work of Liu et al. [2008], the local global algorithm finds an embedding that minimizes:

$$E(u, L) = \sum_{f=1}^{|F|} A_f \|J_f(u) - L_f\|_F^2,$$

where u is the embedding of the mesh in 2D, and L_f are 2×2 matrices constrained to a class of matrices based on the desired parametrization type, i.e. conformal or rigid. Further, $f \in F$ are the faces of the mesh to flatten, with A_f the face area, and $J_f(u)$ the Jacobian of the linear mapping of face f.

To find the best u, L we fix L to be the set of matrices which minimize the energy the most for the respective u.

The best fitting similar and rigid L are known from Procrustes analysis, and are provided by the Singular Value Decomposition on $J_f = U_{J_f} \Sigma_{J_f} V_{J_f}^\top$, where U_{J_f}, V_{J_f} are unitary matrices, and Σ_{J_f} is the diagonal matrix of the singular values $diag(\sigma_{1,J_f}, \sigma_{2,J_f})$:

- ARAP: $L_f = U_{J_f} \cdot diag(1, 1) \cdot V_{J_f}^\top$
- ASAP: $L_f = U_{J_f} \cdot diag(\frac{\sigma_{1,J_f}+\sigma_{2,J_f}}{2}, \frac{\sigma_{1,J_f}+\sigma_{2,J_f}}{2}) \cdot V_{J_f}^\top$

Because J_f, L_f have the same singular vectors, the energy also equals the distance of singular values of the matrices:

$$E(u) = \sum_{f=1}^{|F|} A_f \|\Sigma_{J_f} - \Sigma_{L_f}\|_f^2$$

$$E(u) = \sum_{f=1}^{|F|} A_f \left[(\sigma_{1,J_f} - \sigma_{1,L_f})^2 + (\sigma_{2,J_f} - \sigma_{2,L_f})^2 \right],$$

substituting $\sigma_{1,L_f}, \sigma_{2,L_f}$ with the relevant new singular values.

Finding the best fitting L_f that preserves the area, which means its singular values satisfy the relation $\sigma_{1,L_f} = \frac{1}{k}, \sigma_{2,L_f} = k$ for some positive k, requires minimizing an energy which leads to solving a fourth degree polynomial:

$$\min_k E_u(k) = \min_k [(\sigma_{1,J_f} - \frac{1}{k})^2 + (\sigma_{2,J_f} - k)^2]$$

Although it is possible to solve this numerically, this does not lead to a closed form energy function. Instead, we suggest minimizing the Log-Euclidean distance which leads to a simple solution.

$$\begin{aligned}\min_k E_u(k) &= \min_k [(\ln(\sigma_{1,J_f}) - \ln(\frac{1}{k}))^2 + (\ln(\sigma_{2,J_f}) - \ln(k))^2] \\ &= \min_k [(\ln(\sigma_{1,J_f}) + \ln(k))^2 + (\ln(\sigma_{2,J_f}) - ln(k))^2]\end{aligned}$$

To find the minimum we take the derivative $\frac{d}{dk} E_u(k)$:

$$\begin{aligned}\frac{d}{dk} E_u(k) &= \frac{2\left(\ln(\sigma_{1,J_f}) + \ln(k)\right)}{k} - \frac{2\left(\ln(\sigma_{2,J_f}) - \ln(k)\right)}{k} \\ &= \frac{2\left(\ln(\sigma_{1,J_f}) - \ln(\sigma_{2,J_f}) + 2\ln(k)\right)}{k} \\ &= \frac{2\left(\ln(\frac{\sigma_{1,J_f}}{\sigma_{2,J_f}}) + 2\ln(k)\right)}{k}\end{aligned}$$

And solve for $\frac{d}{dk} E_u(k) = 0, \quad k > 0$:

$$\begin{aligned}&\frac{2\left(\ln(\frac{\sigma_{1,J_f}}{\sigma_{2,J_f}}) + 2\ln(k)\right)}{k} = 0 \\ &\ln(\frac{\sigma_{1,J_f}}{\sigma_{2,J_f}}) + 2\ln(k) = 0 \\ &\ln(k) = \frac{1}{2}\ln(\frac{\sigma_{2,J_f}}{\sigma_{1,J_f}}) \\ &k = \sqrt{\frac{\sigma_{2,J_f}}{\sigma_{1,J_f}}} = \frac{\sigma_{2,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}}\end{aligned} \tag{12}$$

Hence our AEAP singular values:

$$\sigma_{1,L_f} = \frac{\sigma_{1,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}}, \quad \sigma_{2,L_f} = \frac{\sigma_{2,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}} \tag{13}$$

To stop the algorithm when it converged we need to compute an energy. We substitute the singular values in the global Euclidean energy with our AEAP singular values $\frac{\sigma_{1,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}}, \frac{\sigma_{2,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}}$ to obtain an energy:

$$\begin{aligned}E &= \sum_{f=1}^{|F|} A_f \left[(\sigma_{1,J_f} - \frac{\sigma_{1,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}})^2 + (\sigma_{2,J_f} - \frac{\sigma_{2,J_f}}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}})^2 \right] \\ &= \sum_{f=1}^{|F|} A_f \left[\left(\frac{\sigma_{1,J_f}(\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}} - 1)}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}} \right)^2 + \left(\frac{\sigma_{2,J_f}(\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}} - 1)}{\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}}} \right)^2 \right] \\ &= \sum_{f=1}^{|F|} A_f \left[\frac{\sigma_{1,J_f}}{\sigma_{2,J_f}} (\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}} - 1)^2 + \frac{\sigma_{2,J_f}}{\sigma_{1,J_f}} (\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}} - 1)^2 \right] \\ &= \sum_{f=1}^{|F|} A_f \left[\left(\frac{\sigma_{1,J_f}}{\sigma_{2,J_f}} + \frac{\sigma_{2,J_f}}{\sigma_{1,J_f}} \right) (\sqrt{\sigma_{1,J_f}\sigma_{2,J_f}} - 1)^2 \right].\end{aligned} \tag{14}$$

We note that since we're using the Log Euclidean objective for the local step, it is not clear that the global energy should be monotonically decreasing, and the convergence of this local/global scheme is not guaranteed. However, we experimentally found that it was indeed monotonically decreasing in our examples, and the embedding converged using this scheme to an to area-preserving map.

6 Computation Times

Table 1 presents a comparison of computation times between edge length squared blending methods and metric blending methods. All experiments were performed in MATLAB on a system equipped

Table 1. Computation times in seconds for various meshes, deformation types and parameterizations.

	Edge Length Blending		Metric Blending		
	BFF	**CETM**	**LSCM**	**LG-ARAP**	**LG-AEAP**
Stage/Mesh	(Fig. 1) Flamingo Low-Distortion #faces = 818				
Preprocess	–		0.00081		
Blending	0.00004		0.00969		
Realization	0.00375	0.02921	0.00665	0.49061	2.6886
Stage/Mesh	(Fig. 5) Colorbar Conformal #faces = 468				
Preprocess	–		0.00055		
Blending	0.00003		0.00577		
Realization	0.00249	0.02049	0.00395	0.08179	12.634
Stage/Mesh	(Fig. 6) Troll Low-Distortion #faces = 1586				
Preprocess	–		0.00113		
Blending	0.00006		0.01835		
Realization	0.00618	0.08014	0.01232	1.0785	28.586
Stage/Mesh	(Fig. 7) Raptor Low-Distortion #faces = 4554				
Preprocess	–		0.00259		
Blending	0.00009		0.05264		
Realization	0.01644	0.45418	0.03365	5.0381	5.0943
Stage/Mesh	(Fig. 7) Cat High-Distortion #faces = 1590				
Preprocess	–		0.00120		
Blending	0.00006		0.02028		
Realization	0.00650	0.09110	0.01517	0.43985	25.443
Stage/Mesh	(Fig. 7) Blue-Monster High-Distortion #faces = 1573				
Preprocess	–		0.00127		
Blending	0.00007		0.01992		
Realization	0.00694	0.08719	0.01301	0.54159	23.129
Stage/Mesh	(Fig. 8) Cactus Low-Distortion #faces = 4982				
Preprocess	–		0.00263		
Blending	0.00010		0.05702		
Realization	0.02034	0.48316	0.03861	2.7470	3.5787
Stage/Mesh	(Fig. 9) Horse High-Distortion #faces = 2670				
Preprocess	–		0.00159		
Blending	0.00008		0.03191		
Realization	0.01016	0.18787	0.02051	0.92910	17.075
Stage/Mesh	(Fig. 10) Giraffe Low-Distortion #faces = 1583				
Preprocess	–		0.00115		
Blending	0.00007		0.02082		
Realization	0.00692	0.08770	0.01335	3.5590	38.033

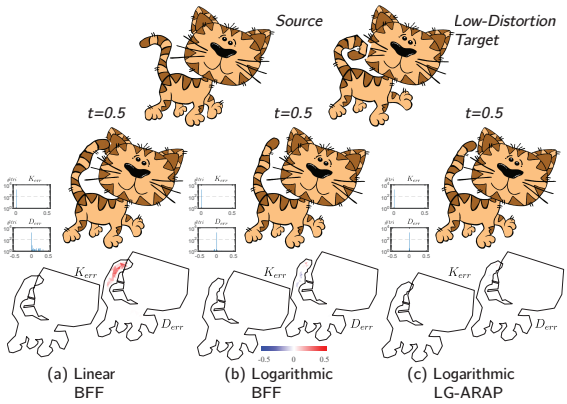

Fig. 1. Interpolation of an As-Killing-As-Possible deformation, with texture and the corresponding conformal and area distortion errors K_{err}, D_{err} plots and histograms. Logarithmic BFF (b) and LG-ARAP (c) bound area distortion better than linear BFF (a).

with an Intel Core i7-10850H CPU @ 2.70GHz. For CETM, the optimization is terminated after 10 iterations, whereas our proposed methods, LG-ARAP and LG-AEAP, employ adaptive stopping criteria based on convergence: either when the energy ratio satisfies $E(\text{iter})/E(\text{iter}-1) \geq 0.999$ or the energy drops below 1×10^{-7}. Typically, LG-ARAP converges within 10 to 50 iterations, while LG-AEAP may require over 400 iterations.

Overall, BFF is approximately twice as fast as LSCM and several orders of magnitude faster than CETM and Local-Global methods. Although CETM and Local-Global exhibit similar per-iteration times to LSCM, their total computation times are higher due to the larger number of iterations required for convergence. As shown in Fig. 1, logarithmic edge length BFF offers a faster alternative to LG-ARAP in low-distortion scenarios, trading off some precision for significantly improved computation time.

References

Renjie Chen, Ofir Weber, Daniel Keren, and Mirela Ben-Chen. 2013. Planar Shape Interpolation with Bounded Distortion. *ACM Trans. Graph.* 32, 4, Article 108 (2013), 12 pages. https://doi.org/10.1145/2461912.2461983

Ligang Liu, Lei Zhang, Yin Xu, Craig Gotsman, and Steven J. Gortler. 2008. A Local/Global Approach to Mesh Parameterization. In *Proceedings of the Symposium on Geometry Processing (SGP '08)*. 1495–1504.

Feng Luo. 2004. Combinatorial Yamabe Flow on Surfaces. *Communications in Contemporary Mathematics* 6, 5 (2004), 765–780. https://doi.org/10.1142/S0219199704001501

Boris Springborn, Peter Schröder, and Ulrich Pinkall. 2008. Conformal Equivalence of Triangle Meshes. *ACM Trans. Graph.* 27, 3 (2008), 11. https://doi.org/10.1145/1360612.1360676