Supplemental Material

To *Implicit Minimal Surfaces for Bijective Correspondences* by Etienne Corman, Yousuf Soliman, Robin Magnet, and Mark Gillespie

A Pseudocode

This supplement provides detailed pseudocode for computing bijective correspondences via minimal surfaces.

Subroutines and quantities not defined in pseudocode are described in the list below.

- a_{ijk} — area of face ijk.
- θ_i^{jk} — corner angle for vertex i in face ijk.
- $\mathrm{d}_1 \in \mathbb{R}^{F\times E}$ — discrete exterior derivative [Desbrun et al. 2005].
- $*_1 \in \mathbb{R}^{E\times E}$ — Hodge star [Desbrun et al. 2005].
- $M_S \in \mathbb{R}^{V\times V}$ — the (scalar) vertex lumped mass matrix on triangle mesh S. Note that this is different from the complex connection mass matrix M_S^∇ computed in Algorithm 7.
- $d_S(p, q), d_S(p, \gamma)$ — the geodesic distance along S from point $p \in S$ to point $q \in S$, or to curve $\gamma \subset S$. (See *e.g.* [Crane et al. 2017].)
- $\textsc{LinearSolve}(\mathsf{A}, \mathsf{b})$ — solves a sparse linear system $\mathsf{Ax} = \mathsf{b}$.
- $\textsc{MinEigenvalue}(\mathsf{A}, \mathsf{B})$ — computes the smallest eigenvalue of the pair A, B, *i.e.* the smallest λ so that exist an x with $\mathsf{Ax} = \lambda\mathsf{Bx}$.
- $\textsc{MinEigenvector}(\mathsf{x} \mapsto \mathsf{Ax}, \mathsf{x} \mapsto \mathsf{Bx})$ — computes the eigenvector of the pair A, B with smallest eigenvalue. For efficiency, we formulate the problem via the operators A, B and avoid assembling the whole product space matrices.
- $\mathrm{LBFGS}((f, \nabla f), x_0)$ — minimize f using LBFGS, starting from initial point x_0, returning optimized point x
- $\textsc{NormalizeToUnitSurfaceArea}(M)$ — scale triangle mesh M so that it has surface area 1.

Product-space matrices. Recall from Section 3.3 that the product mesh $A \times B$ has vertex set $V_{A\times B} = V_A \times V_B$, and so it is convenient to represent discrete sections $z \in \mathbb{C}^{V_{A\times B}}$ by matrices $Z \in \mathbb{C}^{V_A \times V_B}$. Similarly, the product mesh has edge set $E_{A\times B} = E_A \times V_B \cup V_A \times E_B$, so we can represent a connection $r \in \mathbb{C}^{E_{A\times B}}$ by a *pair* of matrices $r^{E,V} \in \mathbb{C}^{E_A \times V_B}$ and $r^{V,E} \in \mathbb{C}^{V_A \times E_B}$, where $r^{E,V}_{e_A,v_B}$ gives the entry for the product-space edge $e_A \times v_B$, and $r^{V,E}_{v_A,e_B}$ gives the entry for the product-space edge $v_A \times e_B$. We often abuse notation and write r_{e_A,v_B} for the entries of the first matrix in $\mathbb{C}^{E_A\times V_B}$, and write r_{v_A,e_B} for the entries of the second matrix in $\mathbb{C}^{V_A\times E_B}$. Finally, the product mesh has face set $F_{A\times B} = F_A \times V_B \cup E_A \times E_B \cup V_A \times F_B$, and thus we write vectors $\Omega \in \mathbb{R}^{F_{A\times B}}$ as triplets of matrices in $\mathbb{R}^{F_A\times V_B}$, *etc.*. We write the ith row of a matrix M as $M_{i,\bullet}$ and the jth column as $M_{\bullet,j}$.

Algorithm 1 $\textsc{SurfaceConnection}(S)$

Input: A triangle meshes $S = (V, E, F)$ with edge lengths ℓ.

Output: A connection $r \in \mathbb{C}^E$ and compatible curvature $\Omega \in \mathbb{R}^F$ such that $\sum_f \Omega_f = 2\pi$

```
$\Omega_{ijk} \leftarrow \frac{1}{2}(\tilde\theta_i^{jk} + \tilde\theta_j^{ki} + \tilde\theta_k^{ij} - \pi)$        ▷for every face $ijk \in F$
Pick an arbitrary face $f_0 \in F$
$L \leftarrow \mathrm{d}_1 *_1^{-1} \mathrm{d}_1^\top$        ▷2-form Laplacian on $S$
$u \leftarrow \textsc{LinearSolve}(L, \Omega - 2\pi\delta_{f_0})$
$r_{ij} \leftarrow \exp(\imath *_{ij}^{-1} (u_{jil} - u_{ijk}))$        ▷for every edge $ij \in E$
return $r, \Omega$
```

Algorithm 2 $\textsc{FindTriangleZero}(ijk, \omega^0, \Omega^0, |z|)$

Input: A triangle ijk with a rotation $\omega^0_{ij} \in [-\pi, \pi)$ per edge ij, the triangle Gaussian curvature $\Omega^0_{ijk} \in [-\pi, \pi)$ and field magnitude $|z_i| \in \mathbb{R}_{>0}$ per vertex i. The triangle must be singular of index ± 1, *i.e.* $\omega^0_{ij} + \omega^0_{jk} + \omega^0_{ki} + \Omega^0_{ijk} = \pm 2\pi$

Output: The barycentric coordinates (b_i, b_j, b_k) of the zero.

```
for $t = 0 \ldots 1$ do        ▷Interpolate from flat to curved triangle
    $\omega_{ij} \leftarrow \omega^0_{ij} + (1-t)\frac{1}{3}\left(\Omega^0_{ijk} - 2\omega^0_{ij} + \omega^0_{jk} + \omega^0_{ki}\right)$
    $\omega_{jk} \leftarrow \omega^0_{jk} + (1-t)\frac{1}{3}\left(\Omega^0_{ijk} + \omega^0_{ij} - 2\omega^0_{jk} + \omega^0_{ki}\right)$
    $\omega_{ki} \leftarrow \omega^0_{ki} + (1-t)\frac{1}{3}\left(\Omega^0_{ijk} + \omega^0_{ij} + \omega^0_{jk} - 2\omega^0_{ki}\right)$
    $\Omega_{ijk} \leftarrow t\Omega^0_{ijk}$
    Find $b_j, b_k$ solution of Equation 28 using Newton method.
return $(1 - b_j - b_k, b_j, b_k)$
```

Algorithm 3 $\textsc{MapVertex}(A, B, r^B, \Omega^B, Z, v_A)$

Input: Triangle meshes $A = (V_A, E_A, F_A)$ and $B = (V_B, E_B, F_B)$, with the connection $r \in \mathbb{C}^{E_B}$ on B, curvature $\Omega \in \mathbb{R}^{F_B}$ on B, a section $Z \in \mathbb{R}^{V_A\times V_B}$ encoding a map from $A \longrightarrow B$, and a vertex $v_A \in V_A$ which we would like to map.

Output: The image of v_A on B, given as a face $ijk \in F_B$ and barycentric coordinates (b_i, b_j, b_k) recording specific point in face ijk that v_A is mapped to.

```
$z^{(v)} \leftarrow Z_{v_A,\bullet}$        ▷Take $v_A$'th row of $Z$ as a section on $B$
$\omega_{ij} \leftarrow \arg\left(\frac{z_j^{(v)}}{r^B_{ij} z^B_i}\right)$        ▷Equation 12 for each $ij \in E_B$
$\mathrm{ind}^z_{ijk} \leftarrow \frac{1}{2\pi}\left(\mathrm{d}_1\omega + \Omega^B\right)_{ijk}$        ▷Equation 13 for each $ijk \in F_B$
$ijk \leftarrow$ first face with $\mathrm{ind}^z_{ijk} \neq 0$
$(b_i, b_j, b_k) \leftarrow \textsc{FindTriangleZero}(ijk, \omega, \Omega^B, |z^{(v)}|)$
return $ijk, (b_i, b_j, b_k)$
```

Algorithm 4 ComputeCorrespondence$(A, B, \varphi, \psi, \{(l_k^A, l_k^B)\}, \{(\gamma_k^A, \gamma_k^B)\})$

Input: Triangle meshes $A = (V_A, E_A, F_A)$ and $B = (V_B, E_B, F_B)$, and optionally: vertex to face maps $\varphi : V_A \to F_B, \psi : V_B \to F_A$, pairs of matching landmark points $\{(l_k^A, l_k^B)\}$ on A and B respectively, and/or pairs of matching landmark curves $\{\gamma_k^A, \gamma_k^B)\}$ on A and B respectively.

Output: A discrete section $z : V_{A\times B} \to \mathbb{C}$ encoding our optimal bijection between A and B

```
▹Normalize inputs
$A \leftarrow$ NormalizeToUnitSurfaceArea$(A)$
$B \leftarrow$ NormalizeToUnitSurfaceArea$(B)$
▹Build connections on A and B (Section 4.1)
$r^A, \Omega^A \leftarrow$ SurfaceConnection$(A)$                ▹Algorithm 1
$r^B, \Omega^B \leftarrow$ SurfaceConnection$(B)$
▹Build FEM Matrices (Section 3.4.3; Algorithm 7)
$L_A^\nabla, M_A^\nabla \leftarrow$ BuildFEMConnectionMatrices$(A, r^A, \Omega^A)$
$L_B^\nabla, M_B^\nabla \leftarrow$ BuildFEMConnectionMatrices$(B, r^B, \Omega^B)$
▹Build pinning potential (Section 4.6)
$V \leftarrow [1] \in \mathbb{R}^{V_A \times V_B}$
$\sigma_A \leftarrow 1, \sigma_B \leftarrow 1$                ▹Landmark penalty (Section 4.6)
if landmark points were provided then
    for $i_A \in V_A, i_B \in V_B$ do
        $V_{i_A,i_B} \leftarrow 1 - \max_k \exp\left(-\frac{1}{2\sigma_A^2} d_A(i_A, l_k^A)^2 - \frac{1}{2\sigma_B^2} d_B(i_B, l_k^B)^2\right)$
if landmark curves were provided then
    for $i_A \in V_A, i_B \in V_B$ do
        $V^C \leftarrow 1 - \max_k \exp\left(-\frac{1}{2\sigma_A^2} d_A(i_A, \gamma_k^A)^2 - \frac{1}{2\sigma_B^2} d_B(i_B, \gamma_k^B)^2\right)$
        $V_{i_A,i_B} \leftarrow \min(V_{i_A,i_B}, V^C)$
▹Find initial section $z_0$
if $\varphi$ and $\psi$ were provided then
    $z_0 \leftarrow$ MapInitialize$(A, B, r^A, r^B, \Omega^A, \Omega^B, \varphi, \psi)$ ▹Algorithm 5
else
    $z_0 \leftarrow$ RandomComplexMatrix$(|V_A|, |V_B|)$
▹Set Ginzburg-Landau parameter based on eigenvalues of A and B (Section 4.2)
$\lambda \leftarrow 100($MinEigenvalue$(L_A^\nabla, M_A^\nabla)+$MinEigenvalue$(L_B^\nabla, M_B^\nabla))$
▹Optimize Ginzburg-Landau energy (Algorithm 6)
$z \leftarrow$ LBFGS$(z \mapsto$ GinzburgLandau$(A, L_A^\nabla, M_A^\nabla, B, L_B^\nabla, M_B^\nabla, z, \lambda, V), z_0)$
return $z$
```

Algorithm 5 MapInitialize$(A, B, r^A, r^B, \Omega^A, \Omega^B, \varphi, \psi)$

Input: Triangle meshes $A = (V_A, E_A, F_A)$ and $B = (V_B, E_B, F_B)$ with edge lengths ℓ_A, ℓ_B, the surface mesh connections r^A, r^B with curvatures $\Omega^A \in \Omega^2(A)$ and $\Omega^B \in \Omega^2(B)$, and vertex to face maps $\varphi : V_A \to F_B$, $\psi : V_B \to F_A$

Output: A discrete section $z : V_{A\times B} \to \mathbb{C}$ approximating the graphs of φ and ψ

```
▹First, build a connection $r^{\varphi,\psi}$ on the product space
$L_A \leftarrow \mathrm{d}_1^A (*_1^A)^{-1} (\mathrm{d}_1^A)^\top$                ▹2-form Laplacian on A
$L_B \leftarrow \mathrm{d}_1^B (*_1^B)^{-1} (\mathrm{d}_1^B)^\top$                ▹2-form Laplacian on B
for $v_A \in V_A$ do ▹Rig connection on B-slices to map $v_A$ to $\varphi(v_A)$
    $\widetilde{\Omega}_{v_A,\bullet} \leftarrow 2\pi\delta_{\varphi(v_A)}$                ▹target slice curvature
    $u \leftarrow$ LinearSolve$(L_B, \Omega^B - \widetilde{\Omega}_{v_A,\bullet})$
    $\rho \leftarrow (*_1^B)^{-1} (\mathrm{d}_1^B)^\top u$
    $r^{\varphi,\psi}_{(v_A,ij)} \leftarrow \exp(\imath\, \rho_{ij})\, r^B_{ij}$                ▹for every edge $ij \in E_B$
for $v_B \in V_B$ do ▹Rig connection on A-slices to map $v_B$ to $\psi(v_B)$
    $\widetilde{\Omega}_{\bullet,v_B} \leftarrow 2\pi\delta_{\psi(v_B)}$                ▹target slice curvature
    $u \leftarrow$ LinearSolve$(L_A, \Omega^A - \widetilde{\Omega}_{\bullet,v_B})$
    $\rho \leftarrow (*_1^A)^{-1} (\mathrm{d}_1^A)^\top u$
    $r^{\varphi,\psi}_{(v_B,ij)} \leftarrow \exp(\imath\, \rho_{ij})\, r^A_{ij}$                ▹for every edge $ij \in E_A$
return MinEigenvector(
    $Z \mapsto$ SlicewiseConnectionLaplacian$(A, B, r^{\varphi,\psi}, \widetilde{\Omega}, Z)$,
    $Z \mapsto$ ApplyMassMatrix$(A, B, Z))$
```

Algorithm 6 GinzburgLandau$(A, L_A^\nabla, M_A^\nabla, B, L_B^\nabla, M_B^\nabla, z, \lambda, V)$

Input: Triangle meshes $A = (V_A, E_A, F_A)$ and $B = (V_B, E_B, F_B)$ with connection Laplacians $L_A^\nabla \in \mathbb{C}^{V_A \times V_A}$, $L_B^\nabla \in \mathbb{C}^{V_B \times V_B}$, scalar mass matrices $M_A \in \mathbb{R}^{V_A \times V_A}$, $M_B \in \mathbb{R}^{V_B \times V_B}$, as well as the current section $z \in \mathbb{C}^{V_A \times V_B}$, the Ginzburg-Landau parameter $\lambda \in \mathbb{R}_{>0}$, and the pinning potential $V \in \mathbb{R}^{V_A \times V_B}$.

Output: The Ginzburg-Landau energy $\mathcal{GL}_\lambda$ and its gradient $\nabla_z \mathcal{GL}_\lambda$

```
$U \leftarrow 0 \in \mathbb{R}^{V_A \times V_B}$
for $i \in V_A, j \in V_B$ do
    $U_{i,j} \leftarrow |z_{i,j}|^2 - V_{i,j}$
$\mathcal{GL}_\lambda \leftarrow \frac{1}{2} \mathrm{Re}\,\mathrm{tr}\left[z^\dagger \left(L_A^\nabla z \left(M^\nabla B\right)^\top + M_A^\nabla z \left(L_B^\nabla\right)^\top\right)\right]$                ▹Eq.26
        $+ \frac{\lambda}{4} \mathrm{tr}\left[U^\top M_A U M_B^\top\right]$ ▹using real mass matrices $M_A, M_B$
$\nabla_z \mathcal{GL}_\lambda \leftarrow L_A^\nabla z \left(M_B^\nabla\right)^\top + M_A^\nabla z \left(L_B^\nabla\right)^\top$                ▹Eq.27
        $+ \lambda \left(M_A U M_B^\top\right) \odot z$ ▹element-wise product
return $\mathcal{GL}_\lambda, \nabla_z \mathcal{GL}_\lambda$
```

Algorithm 7 BuildFEMConnectionMatrices(A, r, Ω)

Input: A triangle mesh $S = (V, E, F)$ with connection $r \in \mathbb{C}^E$ and curvature $\Omega \in \mathbb{R}^F$.

Output: The connection Laplacian L^∇

```
$L^\nabla, M^\nabla \longleftarrow 0 \in \mathbb{C}^{V \times V}, 0 \in \mathbb{R}^{V \times V}$
for corner ${}_i^{jk} \in S$ do
    ▷ Laplace matrix
    $\alpha \leftarrow (\ell_{ij}^2 - \ell_{jk}^2 + \ell_{ki}^2)/2$        ▷ $\alpha = \langle p_j - p_i, p_k - p_i \rangle$
    $w_L \leftarrow \frac{\overline{r}_{jk}}{a_{ijk}} * \left[\left(\ell_{ij}^2 + \ell_{ki}^2\right) * f_1(\Omega_{ijk}) + \alpha\, f_2(\Omega_{ijk})\right]$   ▷ Algs. 9,10
    $L_{jk}^\nabla += w_L, \quad L_{kj}^\nabla += \overline{w_L}, \quad L_{ii}^\nabla += \frac{1}{4a_{ijk}}\left(\ell_{jk}^2 + \Omega_{ijk}^2 \frac{\ell_{ij}^2 + \alpha + \ell_{ki}^2}{90}\right)$
    ▷ Mass matrix
    $w_M \leftarrow a_{ijk}\, \overline{r}_{jk}\, f_0(\Omega_{ijk})$        ▷ Alg. 8
    $M_{jk}^\nabla += w_M, \quad M_{kj}^\nabla += \overline{w_M}, \quad M_{ii}^\nabla += \frac{1}{6} a_{ijk}$
return $L^\nabla, M^\nabla$
```

Algorithm 8 $f_0(s)$ ▷ *Helper for* BuildFEMConnectionMatrices

Input: A real number $s \in \mathbb{R}$.

Output: The value $f_0(s)$ used in Knöppel et al. [2013, Eq.17].

```
▷ The function $f_0(s)$ has a removable singularity at $s = 0$. One can use the Chebyshev expansion provided by Knöppel et al., or the simple Taylor expansion given here
if $|s| < \frac{1}{10}$ then
    return $\frac{1}{12} - \frac{s^2}{360} + \frac{s^4}{20160} + \imath\left(\frac{s}{60} - \frac{s^3}{2520} + \frac{s^5}{181440}\right)$
else
    return $\frac{1}{3s^4}\left(-6 - 6\imath s + 3s^2 \imath s^3 + 6e^{\imath s}\right)$
```

Algorithm 9 $f_1(s)$ ▷ *Helper for* BuildFEMConnectionMatrices

Input: A real number $s \in \mathbb{R}$.

Output: The value $f_1(s)$ defined by Knöppel et al. [2013, §6.1.1].

```
if $|s| < \frac{1}{10}$ then        ▷ Taylor expansion to handle singularity
    return $\frac{s^2}{120} - \frac{s^4}{2688} + \frac{s^6}{129600} + \imath\left(-\frac{s}{24} + \frac{s^3}{504} - \frac{s^5}{17280}\right)$
else
    return $\frac{1}{s^4}\left(3 + \imath s + \frac{s^4}{24} - \imath\frac{s^5}{60} + (-3 + 2\imath s + \frac{s^2}{2})e^{\imath s}\right)$
```

Algorithm 10 $f_2(s)$ ▷ *Helper for* BuildFEMConnectionMatrices

Input: A real number $s \in \mathbb{R}$.

Output: The value $f_2(s)$ defined by Knöppel et al. [2013, §6.1.1].

```
if $|s| < \frac{1}{10}$ then        ▷ Taylor expansion to handle singularity
    return $-\frac{1}{4} + \frac{s^2}{45} - \frac{s^4}{1120} + \frac{s^6}{56700} + \imath\left(-\frac{s}{24} + \frac{5s^3}{1008} - \frac{7s^5}{51840}\right)$
else
    return $\frac{1}{s^4}\left(4 + \imath s - \imath\frac{s^3}{6} - \frac{s^4}{12} + \imath\frac{s^5}{30} + (-4 + 3\imath s + s^2)e^{\imath s}\right)$
```

Algorithm 11 SlicewiseConnectionLaplacian(A, B, r, Ω, Z)

Input: Triangle meshes $A = (V_A, E_A, F_A)$, $B = (V_B, E_B, F_B)$, a product space connection $r \in \mathbb{C}^{E_{A\times B}}$ with curvature $\Omega \in \mathbb{R}^{F_{A\times B}}$, and a section $Z \in \mathbb{C}^{V_A \times V_B}$

Output: Applies the product space connection Laplacian (Equation 34) for connection r to the section Z to obtain a new section Z'.

```
for $v \in V_A, w \in V_B$ do
    $L_B^{\nabla,v}, \_ \leftarrow$ BuildFEMConnectionMatrices$(B, r_{v,\bullet}, \Omega_{v,\bullet})$
    $L_A^{\nabla,w}, \_ \leftarrow$ BuildFEMConnectionMatrices$(A, r_{\bullet,w}, \Omega_{\bullet,w})$
    $Z'_{v,w} \leftarrow (M_A)_{v,v}\left(L_B^{\nabla,v} Z^\top\right)_{w,v} + (M_B)_{w,w}\left(L_A^{\nabla,w} Z\right)_{v,w}$
return $Z'$
```

Algorithm 12 ApplyMassMatrix(A, B, Z)

Input: Triangle meshes $A = (V_A, E_A, F_A)$, $B = (V_B, E_B, F_B)$, and a section $Z \in \mathbb{C}^{V_A \times V_B}$

Output: Applies the product space mass matrix to the section Z.

```
return $M_A Z' M_B^\top$
```

B Parameters and Meshes Statistics

The parameters used for each figure are reported in Table 1.

Table 1. Parameters for each figure: number of variables, Ginzburg-Landau parameter schedule, pinning parameter (if applicable) and usage of intrinsic triangulation.

	$\lvert V_A\rvert \times \lvert V_B\rvert$	$\lambda = t\lambda_0$	σ	iDT
Fig. 1	4643 × 4818	10, 50, 100	×	✓
Fig. 4	1006 × 998	10, 100	×	✓
Fig. 5 *left*	3001 × 3001	10, 100	×	×
Fig. 5 *right*	2999 × 2999	10, 100	×	×
Fig. 12	502 × 2582	100	×	×
Fig. 14	5000 × 4871	100	×	✓
Fig. 15	2485 × 2429	100	1	✓
Fig. 16	3138 × 3146	75	$1/10$	×
Fig. 17	617 × 1338	100	1	✓
Fig. 18 *left*	512 × 1000	100	×	
Fig. 18 *right*	4098 × 2500	100	×	
Fig. 19 *top*	3000 × 3000	100	×	×
Fig. 19 *bottom*	2000 × 2000	100	×	×
Fig. 13	474 × 512	50	1	×
Fig. 20 *left*	1502 × 1477	100, 50	×	✓
Fig. 20 *center*	1502 × 1477	100, 50	×	✓
Fig. 20 *right*	3000 × 3000	100, 50	×	✓
Fig. 22	954 × 921	100	$1/\sqrt{20}$	×
Fig. 21	2252 × 2277	100	×	✓
Fig. 23	3222 × 6121	10, 100	×	×
Fig. 24	4593 × 4017	100	×	×
Fig. 26 *top*	1500 × 1500	100	×	×
Fig. 26 *middle*	1500 × 1500	100	×	×
Fig. 26 *bottom*	1500 × 1500	100	×	×
Fig. 27 *left*	2000 × 2019	100	$1/4$	×
Fig. 27 *middle*	2000 × 3000	100	$1/4$	×
Fig. 27 *right*	2000 × 3000	100	$1/4$	×
Fig. 25	3863 × 3863	25, 75	×	✓