Surface Map Homology Inference

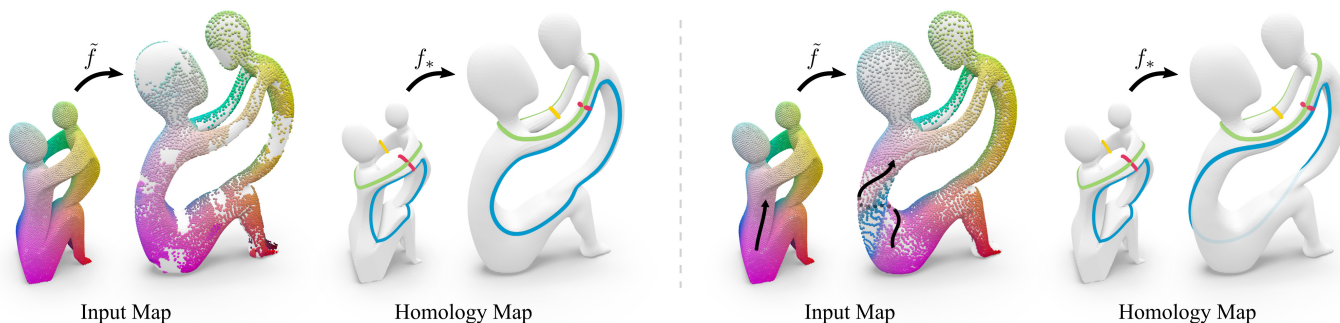J. Born[1] [iD]   P. Schmidt[1] [iD]   M. Campen[2] [iD]   L. Kobbelt[1] [iD]

[1]RWTH Aachen University, Germany   [2]Osnabrück University, Germany

Figure 1: *Left: visualization of a map $\tilde{f}$ between two surfaces. Like many maps used in geometry processing applications, it is not a formal continuous bijection, but a set of sparse point correspondences, with noise, outliers, non-injective overlaps, and gaps. From such a map we infer a homology map f_*, i.e., a topological identification of homology classes, that corresponds to a homeomorphic completion or correction of the imperfect surface map $\tilde{f}$. Homology classes are illustrated by representative cycles. Right: a different input map $\tilde{f}$ that includes a deliberate twist around the waist of the statuette. The inferred homology map f_* correctly captures this twist.*

Abstract

A homeomorphism between two surfaces not only defines a (continuous and bijective) geometric correspondence of points but also (by implication) an identification of topological features, i.e. handles and tunnels, and how the map twists around them. However, in practice, surface maps are often encoded via sparse correspondences or fuzzy representations that merely approximate a homeomorphism and are therefore inherently ambiguous about map topology. In this work, we show a way to infer topological information from an imperfect input map between two shapes. In particular, we compute a homology map, a linear map that transports homology classes of cycles from one surface to the other, subject to a global consistency constraint. Our inference robustly handles imperfect (e.g., partial, sparse, fuzzy, noisy, outlier-ridden, non-injective) input maps and is guaranteed to produce homology maps that are compatible with true homeomorphisms between the input shapes. Homology maps inferred by our method can be directly used to transfer homological information between shapes, or serve as foundation for the construction of a proper homeomorphism guided by the input map, e.g., via compatible surface decomposition.

CCS Concepts
• Computing methodologies → Shape modeling;

1. Introduction

A precise point-wise correspondence between two surfaces is described by a continuous bijective map, i.e. a homeomorphism. Besides the geometric one-to-one relation of surface points, a given homeomorphism determines a specific map topology, an identification of topological features such as handles and tunnels and how often the map twists around each of them. Representations of surface maps used in most geometry processing scenarios, however, are neither truly continuous nor bijective: Many common forms of encoding only describe a finite set of corresponding surface points (e.g. samples from a continuous map; per-vertex images; sparse landmark annotations), or soft, fuzzy, non-sharp correspondences

of larger surface regions. They are thus inherently ambiguous with regard to their map-topological properties. In practice, maps can additionally come with various defects such as noise, outliers, undersampling, or overlaps. Recovering an actual homeomorphism (as assumed by advanced geometry processing and map optimization techniques) from such an imperfect input map requires settling on a map topology which resolves ambiguities and defects in a globally consistent way.

We describe a method to infer a topological correspondence description from a (possibly low-quality) input map. Specifically, we extract a *homology map*, which defines a one-to-one mapping between homology classes (of cycles) on the source and target shape.

While such a homology map does not determine the topology of a homeomorphism entirely in general, it captures the most fundamental topological features, e.g., the identification of corresponding handles and tunnels. Our method addresses the problem that for the above mentioned practical surface maps, which are not true homeomorphisms, this homology map cannot be easily read off unambiguously by simply mapping and comparing representative cycles (homology bases).

We demonstrate the method's support for a large variety of input map representations and its high level of robustness to a wide range of input map imperfections, as commonly encountered in geometry processing applications. The homology map's utility is exemplified in the contexts of surface data transfer in a homologically correct manner and the recovery and construction of true surface homeomorphisms from approximate input.

Contribution. In summary, we introduce a method to infer an unambiguous homology map from an ambiguous surface map. The method exhibits the following properties:

- It supports a large variety of input map types, from dense vertex-to-vertex and vertex-to-triangle maps, over sparse correspondences, to soft or functional maps, and extrinsic registrations.
- It exhibits a favorable level of resilience to a wide range of map imperfections.
- It prevents degeneration of the homology map regardless of input map quality.

Technically, the method's robustness in the face of imperfect maps hinges on three main ingredients:

- Instead of attempting to directly transfer and compare *local* and *high-frequency* information (like surface paths or cycles) via an unreliable map, we carefully choose *global* and *low-frequency* operations, reducing sensitivity to imperfections. Only maximally smooth data is mapped, and compared globally.
- Differential information is mapped effectively without reference to the map's unreliable differential, by encoding it as coordinate-free scalar information.
- Further resilience to low-quality and even misguiding input is achieved by incorporating a global topological consistency constraint, restricting the inference process to the subspace of homology maps that are compatible with homeomorphisms.

Computationally, our inference task in its core boils down to solving an integer quadratic program of modest size, $O(g^2)$ for genus g, which can be efficiently handled by off-the-shelf solvers.

2. Related Work

Homology and cohomology have been considered in a number of works in the field of geometry processing on discrete surfaces. Erickson and Whittlesey [EW05] and Dey et al. [DFW13] describe efficient algorithms for the construction of (greedy) homology bases on surfaces of arbitrary genus. The construction of (canonical) cohomology bases is addressed by Gu and Yau [GY03] and others [PSKG13]. Computational tools built on homology concepts have been used, e.g., to measure topological similarity of curves [CVJ15], for function optimization [PSO18], or surface decomposition [CG19]. We apply them in the field of surface maps.
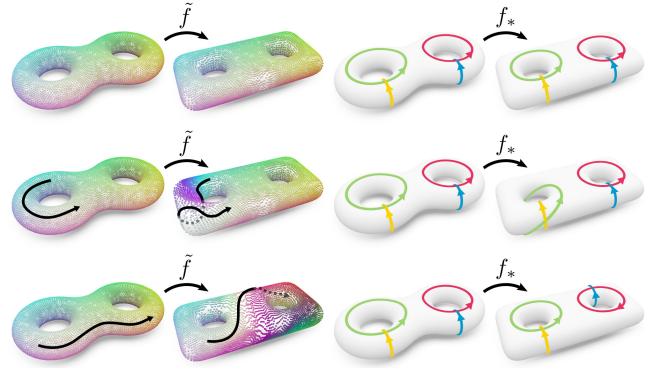
Figure 2: *Left: three densely sampled input maps between the same pair of shapes, describing topologically different maps, as highlighted by black curves and their images. Right: our inferred homology maps capture the implied map topology of each case.*

A number of works consider the completion of sparse surface correspondence information to a continuous bijective map. A common theme is the construction of a structurally compatible surface decomposition into one or multiple disk patches; the map's topological degrees of freedom are settled in this process. Afterwards, per-patch bijections are easily established. The topological degrees of freedom are fixed while building the decomposition cut graph either in a local greedy manner [SAPH04, KS04] or by requiring additional user input [LGQ08, LBG*08, CJGQ05]. A recent non-greedy method [BSK21] is restricted to surfaces of genus zero. These methods are commonly tailored to sparse landmark or curve correspondences, and take this input as hard truth, assuming the absence of outliers or similar imperfections.

Examples for methods that operate on actual surface homeomorphisms (taken as input or constructed along the lines of the above techniques), for instance for the purpose of performing optimization in the space of homeomorphisms, are recent methods for inter-surface map optimization [AL16, SBCK19, SCBK20], compatible re-meshing [YZL*20], or seamless parametrization [APL15].

The large variety of (imperfect) map representations used in geometry processing applications is exemplified by vertex-to-vertex maps [RMC17], vertex-to-surface maps [EHA*19, PBDSH13], sample-to-sample maps [VLB*17, MCSK*17], landmark maps [EEBC20], partial maps [MGP06, BB08], functional maps [OBCS*12], and soft maps [SNB*12].

A number of techniques target the improvement of such maps, by denoising [EBC17], sharpening [RMC17], upsampling [HLC20], or symmetrization [ESBC19]. Nevertheless, they still produce a map of the same type, which remains topologically ambiguous.

3. Background

In this work, our focus lies on the topological properties of surfaces captured by their first homology and cohomology groups. Here, we give a brief introduction to the concepts specific to our application. For a general rigorous treatment, we refer to [Bre93, Hat02, Lee13].
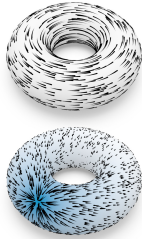
3.1. Homology

In our setting, *homology* describes an equivalence relation of cycles on a surface. On a closed, orientable, two-manifold surface $\mathcal{S}$, a *cycle* c is a set of closed, oriented curves embedded in $\mathcal{S}$. Cycles can be reversed ($-c$) as well as composed via addition: $c + d$ superimposes the cycles c and d (with cancellation where c and d have opposite orientation). A cycle is called *boundary cycle* if it is the oriented boundary of some subregion of $\mathcal{S}$. Two cycles are *homologous*, $c \sim d$, if their difference $c - d$ is a boundary cycle. The *homology class* $[c]$ is an equivalence class of all cycles homologous to c and $[0]$ denotes the class of all boundary cycles. The operation of cycle addition transfers to homology classes via $[c] + [d] = [c+d]$. With this, the set of all homology classes forms a commutative group called the *first homology group* $H_1(\mathcal{S})$.
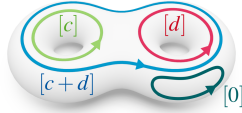
The *algebraic intersection number* $\omega(c, d)$ of a pair of cycles c, d is the sum of their oriented transversal intersections: At each intersection, we count $+1$ if c crosses d left-to-right, or -1 if c crosses d right-to-left. $\omega(c, d)$ is bilinear (w.r.t. cycle addition) and antisymmetric, i.e. $\omega(c, d) = -\omega(d, c)$. It is invariant w.r.t. homologous arguments [FM11, Ch. 1.2.3], hence we can consider it as a bilinear form on homology classes $H_1(\mathcal{S}) \times H_1(\mathcal{S}) \to \mathbb{Z}$ as $\omega([c], [d]) = \omega(c, d)$.

3.2. Cohomology

The related concept of *cohomology* defines an equivalence relation between closed 1-forms on a surface $\mathcal{S}$. A 1-form $x : T\mathcal{S} \to \mathbb{R}$ is *closed* if its exterior derivative vanishes, $dx = 0$ (inset, top). The class of closed 1-forms is closed under negation and addition, i.e., they can be superimposed. A 1-form is *exact* if it is the exterior derivative of some 0-form ϕ, i.e. $x = d\phi$ (inset, bottom). Exact 1-forms are closed ($dx = dd\phi = 0$). Two closed 1-forms x, y are *cohomologous*, $x \sim y$, if their difference $x - y$ is exact. A *cohomology class* $[x]$ is the equivalence class of all closed 1-forms cohomologous to x. As before, we can define an operation $[x] + [y] = [x + y]$ on cohomology classes, allowing us to form the *first cohomology group* $H^1(\mathcal{S})$ of $\mathcal{S}$.

Integration of closed 1-forms along cycles is invariant under homology and cohomology: $\int_{c_1} x_1 = \int_{c_2} x_2$ for any $c_1 \sim c_2$ and $x_1 \sim x_2$. It therefore makes sense to define this operation on classes: $\int_{[c]} [x] = \int_c x$. Due to the linearity of cycle and 1-form addition, this operation defines a non-degenerate bilinear form $H_1(\mathcal{S}) \times H^1(\mathcal{S}) \to \mathbb{R}$ between homology and cohomology classes. If we restrict to the sub-class of closed 1-forms that have integer integrals along all cycles (which is likewise closed under negation and addition), then this bilinear form is onto $\mathbb{Z}$, implying that $H_1(\mathcal{S})$ and $H^1(\mathcal{S})$ are a pair of *dual* spaces, i.e. related by a linear isomorphism [Hat02, Ch. 3.3].

3.3. Homology and Cohomology Bases

For a surface $\mathcal{S}$ of genus g, we can choose a *homology basis* $B_\mathcal{S} = \{s_1, \ldots, s_{2g}\}$, a set of cycles generating the first homology group

$H_1(\mathcal{S})$ (Fig. 3 top left). Any homology class $[c] \in H_1(\mathcal{S})$ is uniquely represented by a linear combination of basis elements

$$[c] = \left[\sum_{i=1}^{2g} s_i h_i \right] = [B_\mathcal{S} h]$$

where $h \in \mathbb{Z}^{2g}$ is a vector of integer coefficients.

A fixed basis $B_\mathcal{S}$ allows an explicit representation of the algebraic intersection form ω via its Gram matrix $\Omega_\mathcal{S} \in \mathbb{Z}^{2g \times 2g}$, with entries $(\Omega_\mathcal{S})_{ij} = \omega(s_i, s_j)$ (Fig. 3 bottom left). Then, for homology classes $[c] = [B_\mathcal{S} h_c]$, $[d] = [B_\mathcal{S} h_d]$, their intersection number is

$$\omega([c], [d]) = \omega([B_\mathcal{S} h_c], [B_\mathcal{S} h_d]) = \langle h_c, h_d \rangle_{\Omega_\mathcal{S}} = h_c^\mathsf{T} \Omega_\mathcal{S} h_d. \quad (1)$$

For a homology basis $B_\mathcal{S}$ on $\mathcal{S}$, a natural choice [GY03, PSKG13] of dual *cohomology basis* $B^\mathcal{S}$, i.e., a generating set of closed 1-forms for $H^1(\mathcal{S})$, is a set of 1-forms $\{s^1, \ldots, s^{2g}\}$ with

$$\int_{s_i} s^j = \delta_{ij} \quad (2)$$

for all i, j (Fig. 3 right).

As before, any cohomology class $[x]$ is represented by a linear combination $[x] = [B^\mathcal{S} h]$ for some coefficients $h \in \mathbb{Z}^{2g}$. In this dual representation, the integral of a cohomology class $[x]$ along some homology class $[c]$ is just the dot product of their coefficients:

$$\int_{[c]} [x] = \int_{[B_\mathcal{S} h_c]} [B^\mathcal{S} h_x] = \langle h_c, h_x \rangle. \quad (3)$$

A basis change between two different homology bases $B_\mathcal{S}$, $B'_\mathcal{S}$ can be described by a transition matrix $M \in \mathbb{Z}^{2g \times 2g}$, mapping coefficients $h \in \mathbb{Z}^{2g}$ w.r.t. $B_\mathcal{S}$ to coefficients $h' = Mh \in \mathbb{Z}^{2g}$ w.r.t. $B'_\mathcal{S}$. Then, duality implies a transition matrix
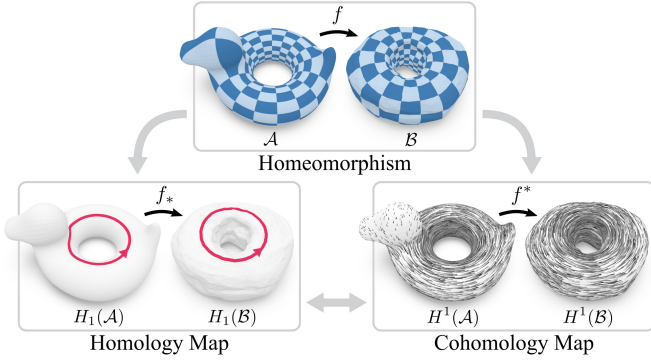
$$\overline{M} = M^{-\mathsf{T}} \quad (4)$$

between coefficients of the dual cohomology bases $B^\mathcal{S}$, $B'^\mathcal{S}$, that preserves the pairing (2).

4. Homology of Surface Maps

Let us now consider two surfaces $\mathcal{A}$ and $\mathcal{B}$, and a homeomorphism (bijective, continuous map) $f : \mathcal{A} \to \mathcal{B}$ between these. This homeomorphism can be characterized topologically via the action of f on the surfaces' homology and cohomology classes.
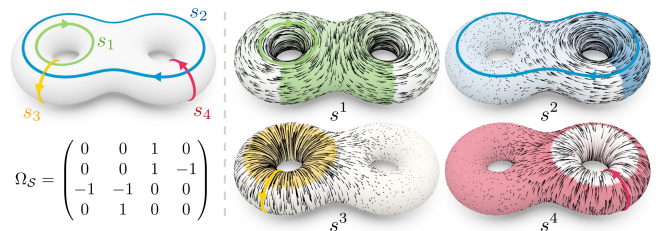
Figure 3: *Left: a homology basis of cycles $B_\mathcal{S} = \{s_1, s_2, s_3, s_4\}$ of a genus 2 surface $\mathcal{S}$. Algebraic intersection numbers $\omega(s_i, s_j)$ of cycles are encoded as coefficients of the matrix $\Omega_\mathcal{S}$. Right: a corresponding dual cohomology basis of closed 1-forms $B^\mathcal{S} = \{s^1, s^2, s^3, s^4\}$, fulfilling $\int_{s_i} s^j = \delta_{ij}$.*

$$\Omega_\mathcal{S} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Figure 4: *A homeomorphism f between surfaces $\mathcal{A}$, $\mathcal{B}$ (top) induces a homology map f_* (bottom left) that maps between homology classes (equivalence classes of cycles) of $\mathcal{A}$ and $\mathcal{B}$, and a corresponding cohomology map f^* (bottom right) that maps between cohomology classes (equivalence classes of closed 1-forms).*

4.1. Induced Homology and Cohomology Maps

The action of f on cycles induces the *homology map $f_* : H_1(\mathcal{A}) \to H_1(\mathcal{B})$* (Fig. 4 bottom left) via

$$f_*([c_{\mathcal{A}}]) = [f(c_{\mathcal{A}})]. \tag{5}$$

This map is a well-defined linear isomorphism between $H_1(\mathcal{A})$ and $H_1(\mathcal{B})$ [Hat02, Ch. 2.1], associating each homology class of $\mathcal{A}$ with a unique homology class of $\mathcal{B}$.

Similarly, it induces the *cohomology map $f^* : H^1(\mathcal{A}) \to H^1(\mathcal{B})$* (Fig. 4 bottom right), defined by

$$f^*([x_{\mathcal{A}}]) = [f(x_{\mathcal{A}})], \tag{6}$$

where f, by slight abuse of notation, here formally denotes the pullback of f^{-1}, mapping 1-forms from $\mathcal{A}$ to $\mathcal{B}$ along f. Also this map is well-defined [Lee13, Ch. 11.2].

4.2. Homology Maps between Bases

In practice, we have independent homology bases $B_{\mathcal{A}}$ on $\mathcal{A}$ and $B_{\mathcal{B}}$ on $\mathcal{B}$ and use coefficient vectors $h_{\mathcal{A}} \in \mathbb{Z}^{2g}$, $h_{\mathcal{B}} \in \mathbb{Z}^{2g}$ to represent homology classes, i.e. $[B_{\mathcal{A}}h_{\mathcal{A}}]$ and $[B_{\mathcal{B}}h_{\mathcal{B}}]$. In this setting, we can express any homology map $f_* : H_1(\mathcal{A}) \to H_1(\mathcal{B})$ explicitly as a map $M : \mathbb{Z}^{2g} \to \mathbb{Z}^{2g}$ between coefficient vectors: For every homology class $[B_{\mathcal{A}}h_{\mathcal{A}}]$, mapped to $\mathcal{B}$ as $f_*([B_{\mathcal{A}}h_{\mathcal{A}}]) = [B_{\mathcal{B}}h_{\mathcal{B}}]$, we have $M(h_{\mathcal{A}}) = h_{\mathcal{B}}$, or equivalently

$$f_*([B_{\mathcal{A}}h_{\mathcal{A}}]) = [B_{\mathcal{B}}M(h_{\mathcal{A}})]. \tag{7}$$

Since f_* is linear, so is M. But in general, not every linear map $M \in \mathbb{Z}^{2g \times 2g}$ represents a homology map induced by some homeomorphism $\mathcal{A} \to \mathcal{B}$. This more restricted class of map representations can be characterized by the succinct constraint

$$\Omega_{\mathcal{A}} = M^{\mathsf{T}}\Omega_{\mathcal{B}}M, \tag{8}$$

where $\Omega_{\mathcal{A}}$ and $\Omega_{\mathcal{B}}$ represent intersection forms (as defined in Sec. 3.3) w.r.t. $B_{\mathcal{A}}$ and $B_{\mathcal{B}}$. This constraint's necessity can be shown based on the fact that homeomorphisms preserve intersections (and therefore intersection numbers) of cycles; see Appendix A for a

derivation. It is also sufficient, as for any matrix M fulfilling (8), there is a homeomorphism $f : \mathcal{A} \to \mathcal{B}$ that induces a homology map f_* represented by M [FM11, Ch. 6.3.2; MP78, Th. 2]. Note that in case $\Omega_{\mathcal{A}} = \Omega_{\mathcal{B}}$ (i.e., $B_{\mathcal{A}}$ and $B_{\mathcal{B}}$ have the same intersection pattern) this constraint defines exactly the class of *symplectic* matrices M. We will refer to (8) as symplectic constraint in general.

4.3. Cohomology Maps between Bases

A homology map representation M can be understood as a change of basis, expressing how f maps basis cycles of $B_{\mathcal{A}}$ onto $\mathcal{B}$, represented w.r.t. $B_{\mathcal{B}}$. The duality of homology and cohomology bases then implies that the dual basis transform (4), i.e. $\overline{M} = M^{-\mathsf{T}}$, represents a corresponding map between cohomology bases $B^{\mathcal{A}}$, $B^{\mathcal{B}}$, i.e., for $h \in \mathbb{Z}^{2g}$, the cohomology map

$$f^*([B^{\mathcal{A}}h]) = [B^{\mathcal{B}}\overline{M}(h)]. \tag{9}$$

In this light, we can express the symplectic constraint (8) on M as an equivalent constraint on $\overline{M}$ as

$$\overline{M}\Omega_{\mathcal{A}}\overline{M}^{\mathsf{T}} = \Omega_{\mathcal{B}}. \tag{10}$$

Note that M and $\overline{M}$ here map between two bases on two surfaces $\mathcal{A}$ and $\mathcal{B}$, whereas in (4) they map between two bases on the same surface $\mathcal{S}$, which is just the special case $\mathcal{A} = \mathcal{B}$.

5. Map Homology Inference

Equipped with this background on homology and cohomology aspects of maps between surfaces, we are now ready to precisely state the problem we address and the algorithmic approach we propose.

5.1. Problem Statement

Our method takes as input:

- A pair of closed surfaces $\mathcal{A}$, $\mathcal{B}$, both of genus g, represented by triangle meshes $(V_{\mathcal{A}}, E_{\mathcal{A}}, F_{\mathcal{A}})$, $(V_{\mathcal{B}}, E_{\mathcal{B}}, F_{\mathcal{B}})$.
- A map $\tilde{f}$ that loosely resembles a homeomorphism $f : \mathcal{A} \to \mathcal{B}$.

We support any map representation that can be used to transfer scalar quantities from $\mathcal{A}$ to $\mathcal{B}$ in some way. This includes sharp dense point-wise maps (e.g. vertex correspondences) and non-sharp representations such as functional maps or soft maps. Sparse maps given by a finite set of corresponding surface points $(p_a, p_b) \in \mathcal{A} \times \mathcal{B}$, e.g. samples or landmarks, can be used as well via interpolation (Sec. 5.5).

Homology bases $B_{\mathcal{A}}$ and $B_{\mathcal{B}}$ of $\mathcal{A}$ and $\mathcal{B}$, represented by $2g$ cycles of oriented edges per mesh, may be provided as input or be generated automatically using [EW05].

From this input, we infer a homology map $f_* : H_1(\mathcal{A}) \to H_1(\mathcal{B})$, represented by a matrix $M \in \mathbb{Z}^{2g \times 2g}$ that maps between $B_{\mathcal{A}}$ and $B_{\mathcal{B}}$ such that

- the inference of f_* is guided by the input map $\tilde{f}$;
- there is a true homeomorphism $f : \mathcal{A} \to \mathcal{B}$ that induces f_*.

In particular, if the imperfect input map $\tilde{f}$ is close to a homeomorphism f, we expect the homology map inferred from $\tilde{f}$ to be the one induced by f. In any case, no matter how bad the input, we ensure that the homology map will never degenerate topologically.
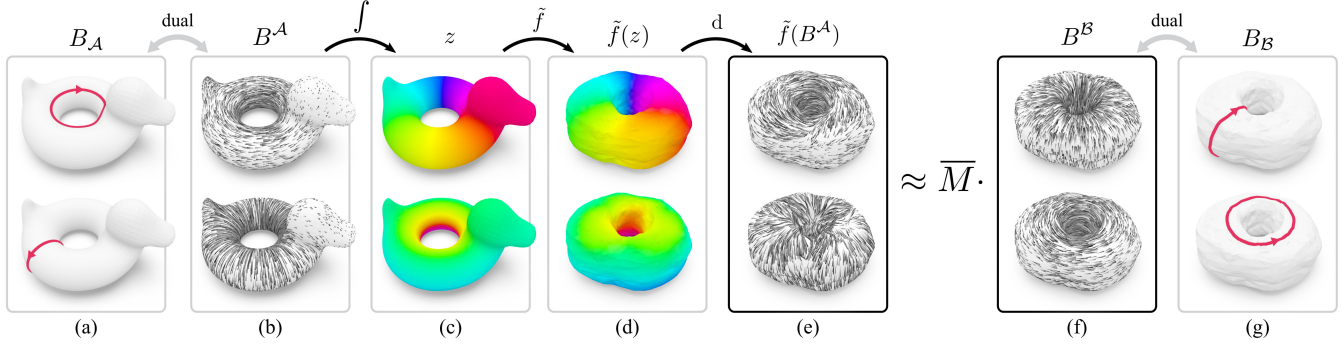
Figure 5: *From a homology basis of surface $\mathcal{A}$ (a), we compute a dual cohomology basis of closed 1-forms (b). We encode these 1-forms on $\mathcal{A}$ as periodic scalar fields z (c) which are then transferred via the given surface map $\tilde{f}$ onto $\mathcal{B}$ (d), where we differentiate to obtain 1-forms again (e). We find an integer matrix $\overline{M}$ that best matches the transferred 1-forms against combinations of cohomology basis representatives on $\mathcal{B}$ (f), subject to symplectic constraint (10) for topological sanity. This cohomology map $\overline{M}$ implies a corresponding homology map M between the homology bases shown in (a) and (g).*

5.2. Algorithm Overview

Since $\tilde{f}$ is sparse and possibly non-injective, we cannot use it directly to map cycles (i.e., homology representatives) from $\mathcal{A}$ to $\mathcal{B}$, which would immediately allow to determine the matrix M of the desired induced homology map f_*. One could try mapping some points of a cycle from $\mathcal{A}$ to $\mathcal{B}$, to the extent $\tilde{f}$ permits, and reconstruct a cycle from the images on $\mathcal{B}$. A major downside of such a *local* approach is its sensitivity to local imperfections such as outliers or noise in the map and consequently its heavy dependence on the choice of cycle. Our aim is to use a *global* approach.

To this end, instead of local high-frequency information, we map global low-frequency information via $\tilde{f}$. Concretely, we transport maximally smooth closed 1-forms (acting as cohomology representatives). As the input map $\tilde{f}$ cannot be assumed to be differentiable (as required to directly map differential information like 1-forms), we take a detour via an encoding as (periodic) scalar fields—that are easily mapped by a wide variety of types of commonly used imperfect maps.

Algorithmically, our approach can be summarized as follows (see Fig. 5 for a visual overview):

1. create a cohomology basis $B^{\mathcal{A}}$ of $2g$ smooth 1-forms (Sec. 5.3)
2. encode $B^{\mathcal{A}}$ as periodic scalar potentials (Sec. 5.4)
3. map them to $\mathcal{B}$ using $\tilde{f}$ with interpolation (Sec. 5.5)
4. decode the mapped potentials to yield 1-forms on $\mathcal{B}$ (Sec. 5.4)
5. collectively match them against a canonical representation of $B^{\mathcal{B}}$, subject to symplectic constraint (10), to infer a cohomology map f^*, represented by a matrix $\overline{M}$ (Sec. 5.6)

Finally, we convert to $M = \overline{M}^{-\top}$ to obtain a representation of the desired homology map f_*, compatible with some homeomorphism f as a consequence of constraint (10), equivalent to (8).

5.3. Harmonic Cohomology Representatives

By Hodge theory [War83, Ch. 6], for each cohomology class $[x]$ there exists a unique representative closed 1-form x that is *harmonic*, i.e. $\Delta x = 0$, thus as smooth as possible. We adopt this choice to construct canonical representatives on $\mathcal{A}$ and $\mathcal{B}$, see Fig. 5b.

For each cycle a_j of our (input) homology basis $B_{\mathcal{A}}$, we compute a corresponding dual cohomology representative $a^j \in B^{\mathcal{A}}$ that is the unique 1-form that fulfills the three constituting properties

$$\mathrm{d}a^j = 0 \qquad (a^j \text{ closed}), \qquad (11)$$

$$\Delta a^j = 0 \qquad (a^j \text{ harmonic}), \qquad (12)$$

$$\textstyle\int_{a_i} a^j = \delta_{ij} \text{ for all } i \qquad (a^j \text{ dual to } a_j). \qquad (13)$$

In our discrete setting, we encode each 1-form as a column vector $a^j = \mathbb{R}^{|E_{\mathcal{A}}|}$ of scalar values assigned to oriented mesh edges. Then, as described by [GY03], properties (11)–(13) discretize to a linear system of equations, which can be solved simultaneously for all a^j. The result is a set of discrete harmonic cohomology generators, encoded as columns of a matrix $B^{\mathcal{A}} \in \mathbb{R}^{|E_{\mathcal{A}}| \times 2g}$. We repeat the same process on $\mathcal{B}$ to compute $B^{\mathcal{B}} \in \mathbb{R}^{|E_{\mathcal{B}}| \times 2g}$. Note that (11) and (13) imply that these 1-forms are *integral*, i.e., they have integer integrals along all cycles (cf. Section 3.2), in particular 1 along respective dual basis cycles.

5.4. Periodic Encoding

Encoding. Given an integral closed 1-form a^j, we compute a real-valued scalar potential $\phi^j : V \to \mathbb{R}$, defined on vertices, by integrating a^j along edges. Starting at some root vertex, we propagate ϕ^j by summing up integrated values of a^j along an arbitrary spanning tree. We then define a complex field $z^j : V \to \mathbb{C}$ as $z^j = \mathrm{e}^{\phi^j 2\pi \mathrm{i}}$, see Fig. 5c, i.e., we use ϕ^j as the argument of a unit complex number.

While ϕ^j depends on the choice of tree and has discontinuities (across the tree's cut locus), z^j is *continuous* and *unique*, i.e., independent of the tree and its root (up to a constant phase shift). This is due to the fact that a^j is closed and integral, implying that ϕ^j has integer jumps at all discontinuities (see Appendix B).

Decoding. A 1-form $\hat{a}^j$ is easily recovered from the complex field z^j: On each edge $e_{pq} \in E$, we compute the value of $\hat{a}^j$ as the smallest angle difference between the arguments of z_p^j and z_q^j:

$$\hat{a}_{pq}^j = \frac{1}{2\pi} \arg\left(z_q^j / z_p^j\right). \qquad (14)$$
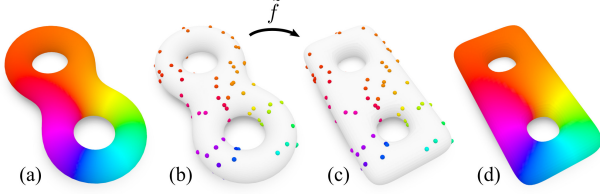
Figure 6: *Transport of a smooth periodic potential under a sparse map $\tilde{f}$ between surfaces: Samples from a complex field (a) on $\mathcal{A}$ are mapped via point-to-point correspondences (b-c) and harmonically extended to a dense, smooth field on $\mathcal{B}$ via interpolation (d).*

where $\arg(z) \in [-\pi, \pi]$ is the argument of $z \in \mathbb{C}$. Note that the recovery is exact ($\hat{a}^j = a^j$) as long as $|a_{pq}^j| < \frac{1}{2}$ for each edge e_{pq}.

5.5. Periodic Potential Mapping

If the input map representation $\tilde{f}$ directly allows the transfer of scalar fields from $V_{\mathcal{A}}$ to $V_{\mathcal{B}}$, we can immediately apply it to transfer the periodic potential z^j of each cohomology representative a^j from $\mathcal{A}$ to $\mathcal{B}$ as $\tilde{f}(z^j) \in V_{\mathcal{B}} \to \mathbb{C}$, see Fig. 5d. Otherwise, if $\tilde{f}$ only defines sparse correspondences $(p_a, p_b) \in \mathcal{A} \times \mathcal{B}$, we define $\tilde{f}(z^j)$ on $\mathcal{B}$ as a field $\tilde{z}^j$ that smoothly interpolates sparse mapped values (Fig. 6). One simple way is to obtain $\tilde{z}^j$ as a least-squares minimizer of a set of interpolation and smoothness terms: For each correspondence (p_a, p_b), we define a fitting term $\|\tilde{z}^j(p_b) - z^j(p_a)\|^2$ (using barycentric interpolation if p_b lies inside a face). We add the harmonicity term $\|\Delta \tilde{z}^j\|^2$ as a regularizer with a small weight, 10^{-3} in our experiments.

We then decode the periodic potential $\tilde{f}(z^j)$ for each j and denote the resulting set of 1-forms by $\tilde{f}(B^{\mathcal{A}}) \in \mathbb{R}^{|E_{\mathcal{B}}| \times 2g}$, see Fig. 5e. While for maps of very low quality or high distortion and sparsity the period ambiguity of (14) might locally be resolved in a way that does match the intent, the global nature of the following procedure gracefully deals with such sporadic effects.

5.6. Symplectic Cohomology Matching

We can now infer a cohomology map f^* (represented by $\overline{M} \in \mathbb{Z}^{2g \times 2g}$) from the above defined action of $\tilde{f}$ on $B^{\mathcal{A}}$. According to (6), the cohomology map f^* is induced by the images of closed 1-forms from $\mathcal{A}$ under $\tilde{f}$. In particular, for any basis element $a^j \in B^{\mathcal{A}}$, it is

$$f^*([a^j]) = [\tilde{f}(a^j)]. \tag{15}$$

In coefficients: $[a^j] = [B^{\mathcal{A}} e_j]$, where $e_j \in \mathbb{Z}^{2g}$ is a standard unit vector. Using (9), we can represent f^* by $\overline{M}$, so (15) becomes

$$[B^{\mathcal{B}} \overline{M} e_j] = [\tilde{f}(a^j)].$$

Therefore, for any basis cycle $b_i \in B_{\mathcal{B}}$, we have

$$\int_{b_i} [B^{\mathcal{B}} \overline{M} e_j] = \int_{b_i} [\tilde{f}(a^j)].$$

With $b_i = B_{\mathcal{B}} e_i$ and (3) the left-hand side becomes $\langle e_i, \overline{M} e_j \rangle = \overline{M}_{ij}$. On the right-hand side, we can drop the brackets and obtain

$$\overline{M}_{ij} = \int_{b_i} \tilde{f}(a^j), \quad \text{or in matrix form:} \quad \overline{M} = B_{\mathcal{B}} \tilde{f}(B^{\mathcal{A}}). \tag{16}$$

Having computed the images $\tilde{f}(a^j)$ of basis 1-forms (Sec. 5.5), we can use (16) to immediately read off the coefficients of $\overline{M}$, representing the induced cohomology map f^*.

For input maps $\tilde{f}$ that are dense and reasonably close to a homeomorphism, (16) will recover its induced cohomology map. However, in the presence of map defects like non-injectivities or undersampling of topological features, it is possible that the interpolated images $\tilde{f}(a^j)$ "snap" into unintended cohomology classes, become linearly dependent, or degenerate to trivial exact 1-forms. By additionally enforcing the symplectic constraint (10), it can be ensured that, regardless of such deficiencies, the inferred cohomology map is compatible with a true homeomorphism. This can be achieved by satisfying (16) in the constrained least squares sense:

$$\min_{\overline{M}} \left\| \overline{M} - B_{\mathcal{B}} \tilde{f}(B^{\mathcal{A}}) \right\|_{\text{F}}^2$$
$$\text{s.t. (10)}, \quad \overline{M} \in \mathbb{Z}^{2g \times 2g}. \tag{17}$$

In practice, we observe that a variation of this formulation is much more robust in particular to low-quality input maps $\tilde{f}$: Instead of comparing cycle integrals of 1-forms (as effectively done in (17)), we infer $\overline{M}$ by directly and globally measuring similarity of these 1-forms. This corresponds to a minor change to (17):

$$\min_{\overline{M}} \left\| B^{\mathcal{B}} \overline{M} - \tilde{f}(B^{\mathcal{A}}) \right\|_{\text{F}}^2$$
$$\text{s.t. (10)}, \quad \overline{M} \in \mathbb{Z}^{2g \times 2g}, \tag{18}$$

i.e. a multiplication with $B^{\mathcal{B}}$ from the left (and using $B^{\mathcal{B}} B_{\mathcal{B}} = I$). This is a quadratic integer problem under quadratic equality constraints. The solution variables are the $(2g)^2$ integer entries of $\overline{M}$. The matrix equality constraint (10) can be encoded coefficient-wise; due to antisymmetry of $\Omega_{\mathcal{A}}$ and $\Omega_{\mathcal{B}}$ (which are constant matrices computed from the intersection patterns of the chosen bases $B_{\mathcal{A}}, B_{\mathcal{B}}$), it suffices to encode the strictly upper triangular part. Due to the small problem size, (18) can be efficiently optimized by off-the-shelf branch-and-bound solvers. We use Gurobi [Gur21] in our experiments. After a cohomology map $\overline{M}$ has been computed, we obtain the desired homology map as $M = \overline{M}^{-\top}$.

Compared to (16), where local map defects can induce unexpected cohomology classes, this formulation is more resilient, as the correct cohomology can still be inferred from correct alignment of 1-forms in reliable regions. The symplectic constraint on $\overline{M}$ further restricts the search space by ruling out maps incompatible with any homeomorphism, globally regularizing the process.

6. Results and Applications

We demonstrate the applicability of our homology inference method for different input map representations (Sec. 6.1) and its robustness to various map defects (Sec. 6.2). The computed homology maps can be used to transfer data in a homologically correct manner (Sec. 6.3) and to topologically control the construction of compatible surface decompositions, e.g. cut graphs or layout embeddings (Secs. 6.4–6.5), and therefore proper homeomorphisms.

To visualize homology classes we will show smooth representative cycles, selected for clarity and visibility.
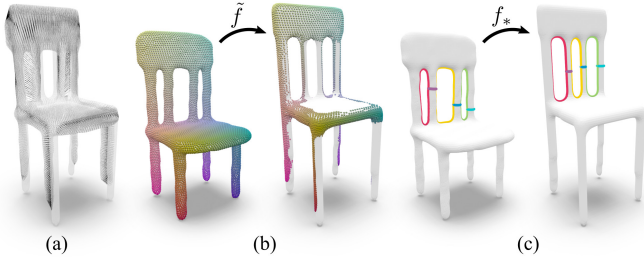
Figure 7: *(a): We establish point-to-point correspondences between two shapes via a rough rigid alignment and closest-point projections. (b): The resulting map $\tilde{f}$ is discontinuous and heavily undersamples or double-covers certain regions of the target surface. (c): Still, we infer the correct homology map f_*.*

6.1. Input Map Representations

We show examples of homology maps inferred from different types of input maps: densely sampled (Fig. 2) or functional maps (Fig. 8), and sparse (Fig. 9) or partial correspondences (Fig. 17).

Due to its high tolerance to map defects, our inference method can be used to quickly generate valid topological correspondences from ad-hoc inputs. One example are vertex-to-surface maps obtained by roughly aligning two shapes in ambient space (e.g. by a rigid transformation) and generating correspondences via simple closest-point projections from one surface to the other (Fig. 7a). While such maps are typically discontinuous and non-injective (Fig. 7b), they often suffice for the inference of the desired homology map (Fig. 7c).

6.2. Robustness

In Fig. 12 we evaluate the robustness of our method with respect to map imperfections such as noise, outliers, and sparsity. Starting from a dense high-quality input map, we progressively degrade the quality of the (sampled) map until the inferred result switches from the original homology map to a different one. First, we simulate noise by displacing map samples on the target surface along random geodesic paths. Our method still tolerates an average geodesic displacement of 24% of the bounding box diagonal and only switches to a different homology at 32%. Second, we examine resilience towards outliers by randomly permuting a subset of samples. Due to the global nature of our objective (18), map inference succeeds even in the presence of 60% outliers. Third, we observe robustness towards very sparse inputs: dropping all but 0.5% (12 samples) of the input map still yields the correct homology in
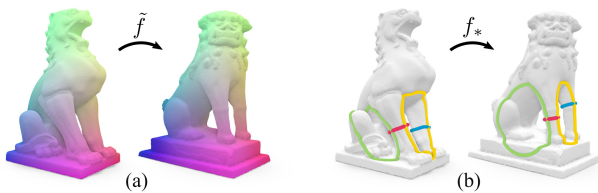
Figure 8: *(a): A functional map $\tilde{f}$ represented by 50×50 coefficients. (b): The homology map f_* obtained from $\tilde{f}$ by our inference.*
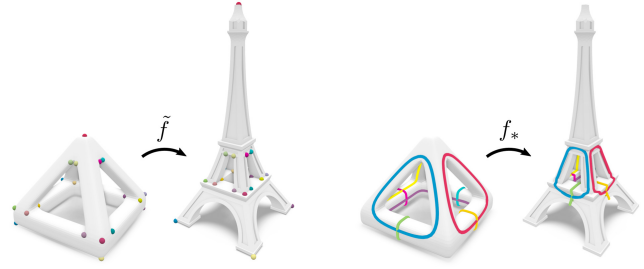
Figure 9: *Homology map f_* computed from an input map $\tilde{f}$ represented by a very sparse set of 29 landmark correspondences on a pair of genus 4 surfaces.*

this experiment. Fig. 13 correspondingly reports the cohomology matrices $\overline{M}$ obtained in these experiments. To demonstrate the effect of the symplectic constraint, we also show the real-valued solutions to an unconstrained version of the optimization problem.

This high resilience to input defects is specific to the similarity-based inference formulation (18). In contrast, the integration-based inference formulation (17) is highly sensitive to mapping and interpolation errors, as demonstrated in Fig. 11.

Besides the ability to cope with various kinds of sampling errors, we also demonstrate robust inference in the presence of large distortions and maps far from isometry in Figs. 9, 10b, 11, and 17.

In Figs. 1, 2, and 16 we emphasize that our method extracts topological information *from an input map* rather than aiming to find a natural matching between two shapes. In all three examples, our inferred homology maps faithfully reproduce the intentional twists indicated by the input maps.

Finally, Fig. 14 displays the successful handling of a challenging combination of higher-genus surfaces and low-quality input map.

6.3. Homological Data Transfer

Induced (co)homology maps can be used to directly transfer data between the (co)homology bases of corresponding objects. One example are harmonic tangent vector fields, which can be expressed as real-valued linear combinations of harmonic fields in a cohomology basis. Fig. 10a demonstrates the transfer of a harmonic vector field from one surface to another via the cohomology map f^*, producing fields with the same flow around corresponding handles.
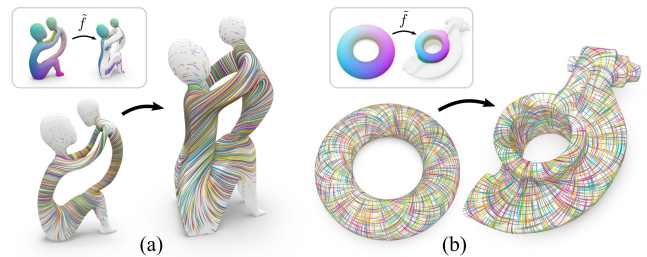

Figure 10: *Left: Transfer of harmonic vector fields (via transfer of real-valued coefficients between cohomology bases). Right: Transfer of direction field turning numbers along homology generators.*
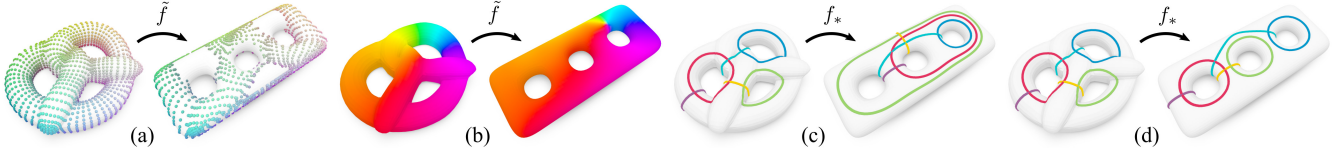
Figure 11: *Due to non-surjectivity and severe misalignment in the input $\tilde{f}$ (a), the interpolated mapping of periodic potentials (b) may yield non-closed 1-forms. In such cases, an integral-based homology inference via (17) may fail easily (c) as it considers integrals along representative paths only; the global similarity-based inference (18) can compensate for such errors and here yields the expected result (d).*
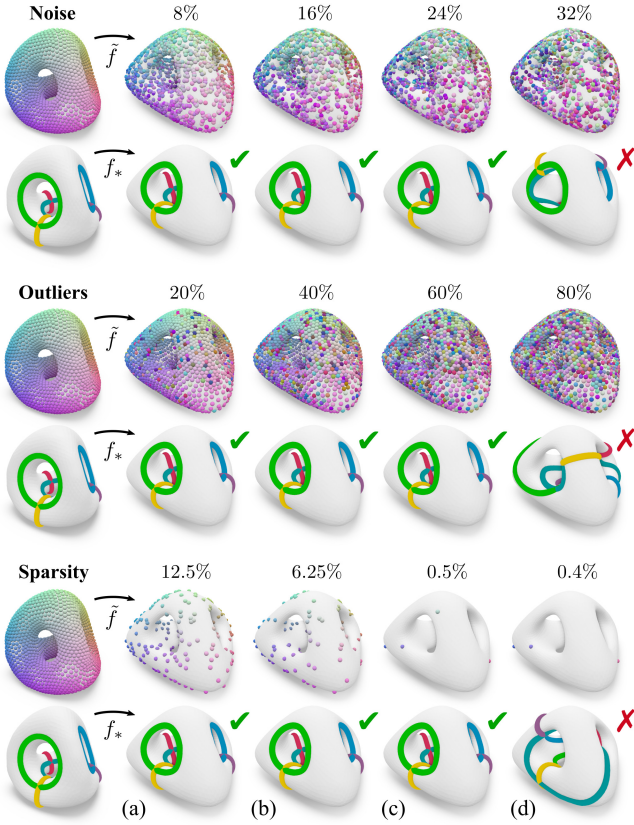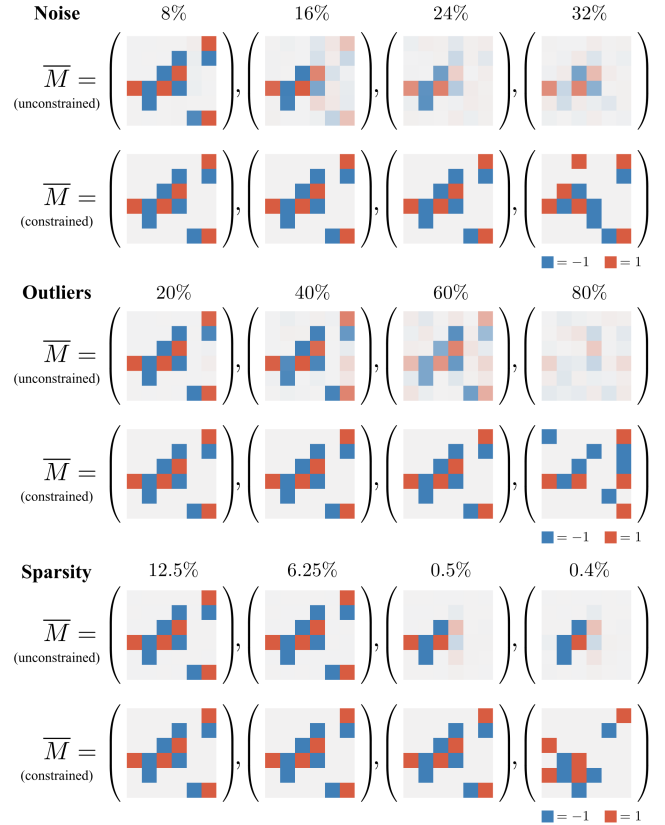

Figure 12: *Robustness towards different input map defects. Noise: samples are randomly displaced by an average geodesic distance. Outliers: a subset of samples is randomly shuffled. Sparsity: only a small subset of samples is kept (11 samples in (c), 10 in (d)). Our method infers the correct homology map in columns (a-c) and only reports a different (but valid) homology in column (d).*

Figure 13: *Cohomology matrices resulting from our optimization. Columns correspond to Fig. 12. Per category: Top rows show the real-valued solutions to the unconstrained problem. Bottom rows show our results, subject to symplectic and integer constraints, which are always valid (co)homology maps. Matrix entries suffering from missing or ambiguous information in the unconstrained case can be faithfully completed via the symplectic constraint.*

Fig. 10b shows a regular cross field on a genus 1 surface. Its global structure can be encoded as turning numbers along basis cycles. Using a homology map f_*, we can transport these turning numbers to reconstruct a field with the same global behavior on another object.

6.4. Compatible Cut Graph Generation

One way to construct a homeomorphism is by cutting two surfaces into disks along compatible cut graphs and overlaying them in a common domain. Fig. 15 shows the construction of cut graphs rooted in a single vertex: Cuts are generated by successively embedding $2g$ non-separating shortest paths that maintain the same cyclic order around the root vertex on both surfaces.

While this process produces cut graphs of compatible connectivity, the topology of embedded cut paths arises from a sequence of greedy decisions, which in general cannot be expected to correspond semantically between the two shapes (Fig. 15a).

However, if semantic correspondence information is given in form of an input map $\tilde{f}$, we can use the induced homology map f_*
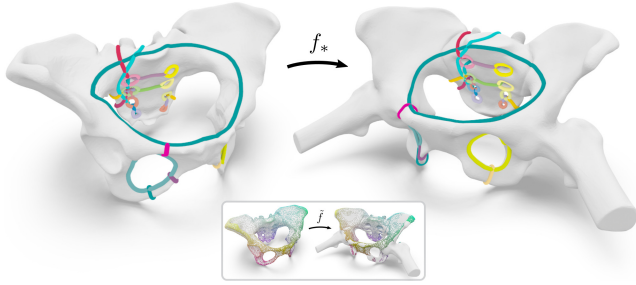
Figure 14: *Homology map f^* inferred from a low-quality map (inset) on a topologically complex surface of genus 12.*

to further constrain the topology of matching paths during cut graph construction: Specifically, for each cut path in homology class $[c]$ on $\mathcal{A}$, we find a matching cut path on $\mathcal{B}$ as a shortest path in homology class $f_*([c])$. Shortest paths under homology constraints can be computed using Dijkstra's algorithm on the universal cover of the target surface $\mathcal{B}$, i.e. on a graph with nodes $V_{\mathcal{B}} \times \mathbb{Z}^{2g}$, and counting intersections with basis cycles $B_{\mathcal{B}}$. Fig. 15b shows the result: Matching cut paths respect topological correspondences specified by the input map and wrap around handles in the same way on both surfaces. Cutting and overlaying both surfaces yields a homeomorphism with the intended map topology (Fig. 15c). It provides a starting point for a subsequent continuous optimization of map geometry – which is necessarily restricted to the initial topology. Fig. 17 shows another example.

Note however, that path homology constraints do not uniquely determine topology: There can be topologically distinct paths in the same homology class (e.g. by including twists around separating cycles, further discussed in Sec. 7). Our homology-constrained shortest-path computation effectively resolves these remaining ambiguities in a greedy manner by picking the shortest option.

6.5. Layout Transfer

Similarly to the above cut graph scenario, a homeomorphism can also be established by embedding a given layout structure into two target surfaces. Again, the topology of the resulting map is decided by a sequence of path embeddings. Choosing shortest paths (as commonly done) thus renders the resulting topology solely dependent on surface geometry.

Our framework adds explicit control over the map topology by supplying additional information via the input map. In Fig. 16 a twist in one of the surfaces is also indicated in the map $\tilde{f}$ and is successfully reproduced in the layout embedding by constraining shortest paths on the target surface to the inferred map homology.

6.6. Computational Cost

We compute harmonic cohomology bases (Sec. 5.3) by solving the linear system (11)–(13), via sparse Cholesky decomposition. In all our examples (meshes of up to 31k vertices) this step never took longer than 1 second.

The remainder of the run time is dominated by solving the integer quadratic program (18) (symplectic cohomology matching,


(a)

(b)

(c)
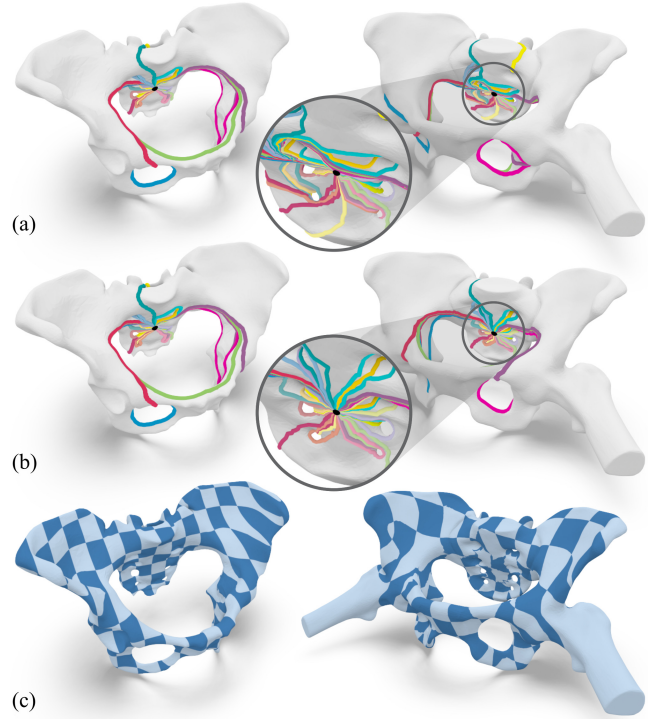
Figure 15: *Generating compatible cut graphs via greedy shortest path embeddings (a) offers little control over the resulting map topology. Given a homology map (Fig. 14), we can constrain paths on both surfaces to matching homology classes (b), here yielding a homeomorphism with the desired topology (c).*

Sec. 5.4), whose number of variables is quadratic in the genus. In practice, we observe that the computational cost depends largely on the quality of the input map. In our experiments, we use a time limit of 60 seconds for the branch-and-bound solver, which is only reached by challenging inputs such as in Fig. 14 (where the returned best-so-far solution is already the semantically correct one), or by the artificially degraded maps in Fig. 12 columns (c) and (d). In all other experiments, the problem was solved in less than 9 seconds, in many cases (Figs. 1, 2, 5, 10, 16) below 1 second.

(a) (b) (c) (d) (e)
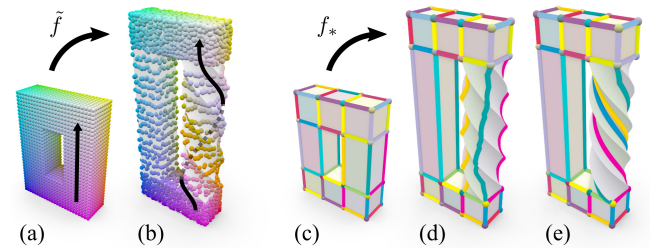
Figure 16: *(a-b): Input map $\tilde{f}$ between surfaces $\mathcal{A}$ and $\mathcal{B}$ indicating a twist. (c): Coarse layout embedded on $\mathcal{A}$. (d): Embedding the same layout structure on $\mathcal{B}$ via shortest paths does not capture the desired twist. (e): Via the inferred homology map f_*, we constrain the resulting layout embedding to the expected topology.*
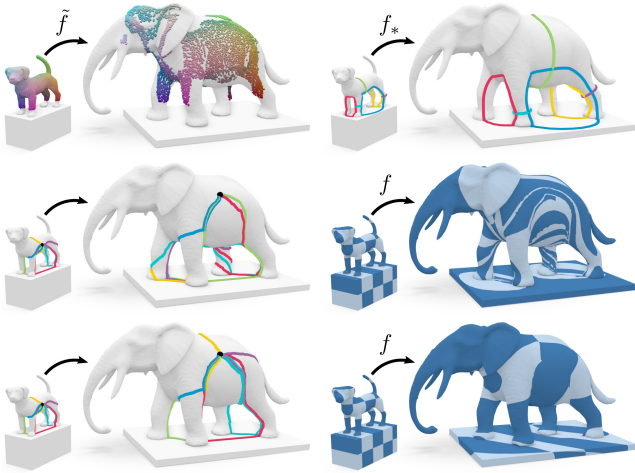
Figure 17: *Inferred homology map (top). Constructing a homeomorphism from compatible cut graphs without homology constraints (middle) is highly susceptible to topological artifacts. With homology constraints (bottom), we obtain a homeomorphism in the desired homotopy class, ready for geometric optimization.*

7. Limitations and Future Work

For very sparse input maps $\tilde{f}$, the transport of 1-forms via interpolation of scalar fields (Sec. 5.5) deteriorates if topological features are undersampled (e.g. entire handles not covered by $\tilde{f}$).

Depending on the number and location of samples, harmonic interpolation may not recover the full periodic signal, as illustrated in the inset. This can result in 1-forms mapped to potentially unintended cohomology classes. Nevertheless, due to the enforced symplectic consistency and the regularizing effect of its global nature, our inference still often yields the intended or expected overall homology map in such cases. Note that one cannot easily argue about the result being *correct* or *incorrect*, because the input map, especially in sparse settings, is inherently ambiguous topologically.

Formulation (18) relies on matching images of harmonic fields from $\mathcal{A}$ under $\tilde{f}$ against harmonic fields on $\mathcal{B}$. This measure is ideal for maps $\tilde{f}$ that preserve harmonicity, such as isometries, because in this case it effectively matches cohomology classes via canonical representatives. While our experiments show that (18) performs robustly even for maps far from isometry, it could be interesting to explore whether an adapted measure that explicitly accounts for non-isometry can offer theoretical or practical benefits.

We have focused our attention on closed surfaces. Extension to surfaces with boundary will be a worthwhile direction for future work. It will require generalized notions and generalized constructions of (co)homology bases. As a special case, this may also allow considering homology with respect to punctured surfaces.

In Secs. 6.4 and 6.5 we have shown how a homology map can be used to constrain the topology of paths, e.g., for the construction of cut graphs, layout embeddings, or homeomorphisms. However, homology does not fully determine path topology in general:

Figure 18: *An input map with a Dehn twist around a separating cycle, a topological feature that cannot be captured by homology. The inferred homology map is identical to the one in Fig. 2 top.*

For surfaces of genus $g \geq 2$, there are homologous paths that are non-*homotopic*, i.e. that cannot be continuously deformed into one another (differing, e.g., by Dehn twists along separating cycles like in the inset) – homology is sometimes said to be a "linear approximation" of homotopy. This gap between homology and homotopy extends to the topology of maps, where it is captured by the quite intricate Torelli group [FM11, Ch. 6; HM12]. Elements of the Torelli group are homologically indistinguishable: The twisted input map in Fig. 18 induces the same homology map as the "identity map" (Fig. 2 top). The direct inference of a homotopy map is therefore an attractive goal. The non-commutativity of the Torelli group and the consequent fact that homotopy classes cannot be dealt with by means of linear algebra poses a key challenge in this regard.

Appendix A: Symplectic Constraint

If two cycles c_1, c_2 on $\mathcal{A}$ have intersection number $\omega_{\mathcal{A}}(c_1, c_2)$ on surface $\mathcal{A}$, their images $f(c_1)$, $f(c_2)$ under a homeomorphism must have the same intersection number on $\mathcal{B}$. This also applies to their homology classes:

$$\omega_{\mathcal{A}}([c_1],[c_2]) = \omega_{\mathcal{B}}([f(c_1)],[f(c_2)]).$$

Following (5), f induces a homology map f_*, i.e.

$$\omega_{\mathcal{A}}([c_1],[c_2]) = \omega_{\mathcal{B}}(f_*([c_1]), f_*([c_2])).$$

Let $B_{\mathcal{A}}$, $B_{\mathcal{B}}$ be homology bases for $\mathcal{A}$, $\mathcal{B}$. Then, we can express homology classes using coefficients $[c_1] = [B_{\mathcal{A}} h_1]$, $[c_2] = [B_{\mathcal{A}} h_2]$:

$$\omega_{\mathcal{A}}([B_{\mathcal{A}} h_1],[B_{\mathcal{A}} h_2]) = \omega_{\mathcal{B}}(f_*([B_{\mathcal{A}} h_1]), f_*([B_{\mathcal{A}} h_2])).$$

In this representation, we can replace f_* by M via (7)

$$\omega_{\mathcal{A}}([B_{\mathcal{A}} h_1],[B_{\mathcal{A}} h_2]) = \omega_{\mathcal{B}}([B_{\mathcal{B}} M(h_1)], [B_{\mathcal{B}} M(h_2)])$$

and compute intersection forms using (1) as

$$\langle h_1, h_2 \rangle_{\Omega_{\mathcal{A}}} = \langle M(h_1), M(h_2) \rangle_{\Omega_{\mathcal{B}}}.$$

Requiring this for all homology classes yields the constraint

$$\Omega_{\mathcal{A}} = M^{\mathsf{T}} \Omega_{\mathcal{B}} M,$$

ensuring that the homology map represented by M preserves intersection numbers between pairs of cycles transported from $\mathcal{A}$ to $\mathcal{B}$.

Appendix B: Periodic Scalar Potentials of Closed 1-Forms

Consider a closed 1-form a^j on $\mathcal{A}$ such that $\int_c a^j \in \mathbb{Z}$ for any cycle c. According to the Poincaré lemma, on any contractible domain, there exists a 0-form (i.e., scalar field) ϕ that has a^j as its gradient, i.e. $a^j = \mathrm{d}\phi$. Since $\mathcal{A}$ is connected, its universal cover C is contractible. We can lift a^j to C as $\hat{a}^j$, which is a closed 1-form on C. Therefore, there exists some $\phi : C \to \mathbb{R}$ on the universal cover with $\mathrm{d}\phi = \hat{a}^j$. Then, ϕ is periodic in the sense that $\phi(c_1) = \phi(c_2) + n$ with $n \in \mathbb{Z}$ for any pair of points $c_1, c_2 \in C$ with identical projections $p(c_1) = p(c_2)$.

Proof: As $p(c_1) = p(c_2)$, there is a path γ connecting c_1 and c_2 in C. Via Stokes' theorem, $\phi(c_2) - \phi(c_1) = \int_{\partial\gamma} \phi = \int_\gamma \mathrm{d}\phi = \int_\gamma \hat{a}^j$. Since $\hat{a}^j$ is a lift of a^j, it is $\int_\gamma \hat{a}^j = \int_{p(\gamma)} a^j$. The projection $p(\gamma)$ onto $\mathcal{A}$ is a closed curve, i.e. a cycle, hence $\int_{p(\gamma)} a^j = n$ is an integer. $\qquad\square$

References

[AL16]   AIGERMAN N., LIPMAN Y.: Hyperbolic orbifold Tutte embeddings. *ACM Trans. Graph. 35*, 6 (2016). 2

[APL15]   AIGERMAN N., PORANNE R., LIPMAN Y.: Seamless surface mappings. *ACM Trans. Graph. 34*, 4 (2015). 2

[BB08]   BRONSTEIN A. M., BRONSTEIN M. M.: Regularized partial matching of rigid shapes. In *European Conference on Computer Vision* (2008), Springer. 2

[Bre93]   BREDON G. E.: *Topology and Geometry*. Springer, 1993. 2

[BSK21]   BORN J., SCHMIDT P., KOBBELT L.: Layout embedding via combinatorial optimization. *Comp. Graph. Forum 40*, 2 (2021). 2

[CG19]   CHEN J., GOPI M.: Geometry aware tori decomposition. *Comp. Graph. Forum 38*, 2 (2019). 2

[CJGQ05]   CARNER C., JIN M., GU X., QIN H.: Topology-driven surface mappings with robust feature alignment. In *VIS 05. IEEE Visualization, 2005.* (2005), IEEE. 2

[CVJ15]   CHAMBERS E. W., VEJDEMO-JOHANSSON M.: Computing minimum area homologies. *Comp. Graph. Forum 34*, 6 (2015). 2

[DFW13]   DEY T. K., FAN F., WANG Y.: An efficient computation of handle and tunnel loops via Reeb graphs. *ACM Trans. Graph. 32*, 4 (2013). 2

[EBC17]   EZUZ D., BEN-CHEN M.: Deblurring and denoising of maps between shapes. *Comp. Graph. Forum 36*, 5 (2017). 2

[EEBC20]   EDELSTEIN M., EZUZ D., BEN-CHEN M.: ENIGMA: Evolutionary non-isometric geometry matching. *ACM Trans. Graph. 39*, 4 (2020). 2

[EHA*19]   EZUZ D., HEEREN B., AZENCOT O., RUMPF M., BEN-CHEN M.: Elastic correspondence between triangle meshes. *Comp. Graph. Forum 38*, 2 (2019). 2

[ESBC19]   EZUZ D., SOLOMON J., BEN-CHEN M.: Reversible harmonic maps between discrete surfaces. *ACM Trans. Graph. 38*, 2 (2019). 2

[EW05]   ERICKSON J., WHITTLESEY K.: Greedy optimal homotopy and homology generators. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms* (2005). 2, 4

[FM11]   FARB B., MARGALIT D.: *A Primer on Mapping Class Groups*. Princeton University Press, 2011. 3, 4, 10

[Gur21]   GUROBI OPTIMIZATION, LLC: Gurobi optimizer reference manual, 2021. URL: http://www.gurobi.com. 6

[GY03]   GU X., YAU S.-T.: Global conformal surface parameterization. In *Proc. Eurographics Symp. on Geometry Processing* (2003). 2, 3, 5

[Hat02]   HATCHER A.: *Algebraic Topology*. Cambridge University Press, 2002. 2, 3, 4

[HLC20]   HOLZSCHUH B., LÄHNER Z., CREMERS D.: Simulated annealing for 3D shape correspondence. In *International Conference on 3D Vision* (2020), vol. 2. 2

[HM12]   HATCHER A., MARGALIT D.: Generating the Torelli group. *L'Enseignement Mathématique 58*, 1 (2012). 10

[KS04]   KRAEVOY V., SHEFFER A.: Cross-parameterization and compatible remeshing of 3D models. *ACM Trans. Graph. 23*, 3 (2004). 2

[LBG*08]   LI X., BAO Y., GUO X., JIN M., GU X., QIN H.: Globally optimal surface mapping for surfaces with arbitrary topology. *IEEE Transactions on Visualization and Computer Graphics 14*, 4 (2008). 2

[Lee13]   LEE J. M.: *Introduction to Smooth Manifolds*. Springer, 2013. 2, 4

[LGQ08]   LI X., GU X., QIN H.: Surface matching using consistent pants decomposition. In *Proceedings of the ACM Symposium on Solid and Physical Modeling* (2008). 2

[MCSK*17]   MANDAD M., COHEN-STEINER D., KOBBELT L., ALLIEZ P., DESBRUN M.: Variance-minimizing transport plans for inter-surface mapping. *ACM Trans. Graph. 36*, 4 (2017). 2

[MGP06]   MITRA N. J., GUIBAS L. J., PAULY M.: Partial and approximate symmetry detection for 3D geometry. *ACM Trans. Graph. 25*, 3 (2006). 2

[MP78]   MEEKS W. H., PATRUSKY J.: Representing homology classes by embedded circles on a compact surface. *Illinois Journal of Mathematics 22*, 2 (1978). 4

[OBCS*12]   OVSJANIKOV M., BEN-CHEN M., SOLOMON J., BUTSCHER A., GUIBAS L.: Functional maps: A flexible representation of maps between shapes. *ACM Trans. Graph. 31*, 4 (2012). 2

[PBDSH13]   PANOZZO D., BARAN I., DIAMANTI O., SORKINE-HORNUNG O.: Weighted averages on surfaces. *ACM Trans. Graph. 32*, 4 (2013). 2

[PSKG13]   PELLIKKA M., SUURINIEMI S., KETTUNEN L., GEUZAINE C.: Homology and cohomology computation in finite element modeling. *SIAM Journal on Scientific Computing 35*, 5 (2013). 2, 3

[PSO18]   POULENARD A., SKRABA P., OVSJANIKOV M.: Topological function optimization for continuous shape matching. In *Comp. Graph. Forum* (2018), vol. 37. 2

[RMC17]   RODOLÀ E., MOELLER M., CREMERS D.: Regularized pointwise map recovery from functional correspondence. *Comp. Graph. Forum 36*, 8 (2017). 2

[SAPH04]   SCHREINER J., ASIRVATHAM A., PRAUN E., HOPPE H.: Inter-surface mapping. *ACM Trans. Graph. 23*, 3 (2004). 2

[SBCK19]   SCHMIDT P., BORN J., CAMPEN M., KOBBELT L.: Distortion-minimizing injective maps between surfaces. *ACM Trans. Graph. 38*, 6 (2019). 2

[SCBK20]   SCHMIDT P., CAMPEN M., BORN J., KOBBELT L.: Inter-surface maps via constant-curvature metrics. *ACM Trans. Graph. 39*, 4 (2020). 2

[SNB*12]   SOLOMON J., NGUYEN A., BUTSCHER A., BEN-CHEN M., GUIBAS L.: Soft maps between surfaces. *Comp. Graph. Forum 31*, 5 (2012). 2

[VLB*17]   VESTNER M., LÄHNER Z., BOYARSKI A., LITANY O., SLOSSBERG R., REMEZ T., RODOLA E., BRONSTEIN A., BRONSTEIN M., KIMMEL R., ET AL.: Efficient deformable shape correspondence via kernel matching. In *IEEE Int. Conf. 3D Vision* (2017). 2

[War83]   WARNER F. W.: *Foundations of Differentiable Manifolds and Lie Groups*. Springer, 1983. 5

[YZL*20]   YANG Y., ZHANG W.-X., LIU Y., LIU L., FU X.-M.: Error-bounded compatible remeshing. *ACM Trans. Graph. 39*, 4 (2020). 2