



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Genus Oblivious Cross Parameterization: Robust Topological Management of Inter-surface Maps

J. C. Bennett, V. Pascucci, K. I. Joy

October 24, 2007

Pacific Graphics
Kaanapali Beach , HI, United States
October 29, 2007 through November 2, 2007

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Genus Oblivious Cross Parameterization: Robust Topological Management of Inter-surface Maps

Janine Bennett^{*†}, Valerio Pascucci^{†*}, and Kenneth Joy^{*}

^{*}University of California, Davis

[†]Lawrence Livermore National Laboratory

Abstract

We consider the problem of generating a map between two triangulated meshes, M and M' , with arbitrary and possibly differing genus. This problem has rarely been tackled in its generality. Early schemes considered only topological spheres. Recent algorithms allow inputs with an arbitrary number of tunnels but require M and M' to have equal genus, mapping tunnel to tunnel. Other schemes which allow more general inputs are not guaranteed to work and the authors do not provide a characterization of the input meshes that can be processed successfully. Moreover, the techniques have difficulty dealing with coarse meshes with many tunnels.

In this paper we present the first robust approach to build a map between two meshes of arbitrary unequal genus. We also provide a simplified method for setting the initial alignment between M and M' , reducing reliance on landmarks and allowing the user to select “landmark tunnels” in addition to the standard landmark vertices. After computing the map, we automatically derive a continuous deformation from M to M' using a variational implicit approach to describe the evolution of non-landmark tunnels.

Overall, we achieve a cross parameterization scheme that is provably robust in the sense that it can map M to M' without constraints on their relative genus or on the density of the triangulation with respect to the number of tunnels. To demonstrate the practical effectiveness of our scheme we provide a number of examples of inter-surface parameterizations between meshes of different genus and shape.

1 Introduction

Cross parameterizations are maps between two input meshes that play a key role in geometry processing algorithms such as morphing, attribute transfer and mesh blending. The key factors that affect the quality of the map include the initial placement of *landmark vertices* (hard con-

straints that the final mapping must maintain), mesh geometry and mesh topology. In this paper we focus on the role of topology in the mapping. Depending on the input model, topological structures may or may not be desired features in the map. For example, model acquisition can introduce small tunnels in the mesh that do not reflect the topology of the original object. We provide a flexible system that minimizes the adverse impact topology can have on the cross parameterization. Our approach has several advantages over previous techniques:

- We provide the first provably robust method to generate mappings between meshes of arbitrary and different genus.
- We extend Morse theory to provide a formal framework to analyze and compare cross parameterization methods.
- We provide a completely parametric system that is not affected by convoluted geometry.
- We propose a novel initial alignment scheme which reduces the total number of landmarks required.

The rest of the paper is outlined as follows. Related work is introduced in §2 and theoretical foundations are discussed in §3. We introduce a formalization in §4 that provides a framework to guarantee robustness and discuss our implementation §5. Results and future work are discussed in §6 and §7.

2 Related Work

Our approach relates to techniques developed in several different fields. We extend Morse theory to provide a formal characterization of parameterization methods. Moreover, we use ideas from genus reduction, previous cross parameterization methods and functional morphing schemes. Here we discuss related work in each of these fields, limiting discussion to topics most relevant to this paper.

Morse Theory. Morse theory characterizes the invariants of a manifold M in terms of the topology of a function f defined on M . The work of [26] details how to minimize the number of critical points of f to describe the genus of M . A skeleton of the shape of M is provided by the Reeb graph which is the contraction of the components of level sets of f to points. In recent years Reeb graphs have been used as a search key in shape databases [18], as well as to characterize complex scientific data [6]. They provide a surface based method for genus reduction [28, 23, 35] that does not require conversion of input models as in volumetric approaches [36, 34, 27]. The notion of persistence was introduced in [10] and is used to rank the importance of topological features [17, 2]. Jacobi sets are defined in [7] and have been used to explore scientific data sets with multiple fields [8]. In this paper we introduce the notion of a Morse field to formally characterize the correctness of a parameterization method.

Mesh Parameterization. Surface parameterizations are bijective maps from a given input mesh to a standard parametric domain that attempt to minimize both conformal and metric distortions. A thorough exposition on the fundamentals of surface parameterization can be found in [12]. The parametric domains of surface mesh parameterization algorithms vary according to the genus of the input mesh. Closed surface genus-0 meshes can be parameterized to spherical domains using extensions of planar techniques [14, 29, 31, 13]. Meshes of arbitrary genus can be mapped to the plane [15, 3] or to a topologically equivalent base domain [20, 21, 30]. Recent work [23] maps meshes of arbitrary genus γ to a series of $\gamma + 1$ spherical domains. One of these domains represents a positive surface while the remaining γ are negative surfaces. The original mesh surface is obtained via boolean difference operations.

Cross Parameterization. Cross parameterizations determine a bijective mapping between a source mesh and a target mesh. Many of the existing approaches [30, 21, 3, 22] require that the source and target be of the same genus. The methods of [30, 21] find compatible base triangulations on the input models. The work presented in [21] is guaranteed to work only on genus zero meshes, while the approach of [30] finds a maximal non-separating cut graph on each input mesh, guaranteeing the success of their algorithm on meshes of higher genus. The work of [3] addresses the homotopy type of a mapping and provides a framework to compute many canonical cuts rather than a single cut graph.

The methods of [5] and [23] allow for maps between meshes of different genus. [5] requires the user to manually specify control meshes with the same number of faces for both the source and target. The recent work of [23] parameterizes meshes of genus γ via a single “positive” spherical

parameterization and γ “negative” spherical parameterizations. The approach works well in many cases, however suffers from several key issues. During genus reduction two boundary loops are computed per tunnel and fine detail in the original mesh may be lost due to the removal of mesh faces between the boundary loops. Pseudo-negative meshes are generated for non-landmark tunnels, however the authors do not provide details on their creation. Furthermore, the algorithm fails on meshes with complicated geometry.

Functional Morphing Techniques. Morphing algorithms can be arranged into two different categories: cross-parametric and functional approaches. Functional approaches do not maintain the mapping between source and target meshes, yet they naturally handle morphing between meshes of unequal genus. The approach of [33] generates a morph sequence between a series of models by computing a distance function for each model. The surface at a given times step is an extremal surface extracted from the weighted average of the functions associated with the input models. The work of [32] computes a variational implicit function in dimension $n + 1$ to solve for the morph sequence between two n -dimensional objects (time is the additional dimension).

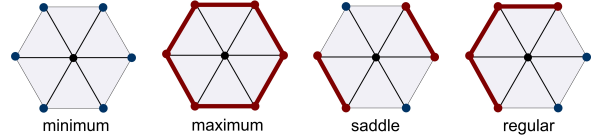


Figure 1. A vertex is classified by its lower link. Here, simplices in the lower link are drawn in red.

3 Foundation

For completeness we briefly introduce mathematical concepts related to Morse theory and variational implicit functions that are used throughout this paper.

Morse theory. A k -simplex is the convex hull of $k + 1$ affinely independent points. A *triangulation* M of a 2-manifold is a set of 0, 1, and 2 simplices commonly called vertices, edges, and faces. A piecewise linear (PL) function f on M is defined by a set of scalar values at the vertices that are extended over the edges and faces of M via linear interpolation. The function f is assumed to be non-degenerate (all function values at vertices are unique) and Simulation of Simplicity [11] guarantees this through symbolic perturbation.

The *star* of a vertex v consists of all simplices in M that contain v and the *link* of a vertex v , denoted $Lk(v)$, consists

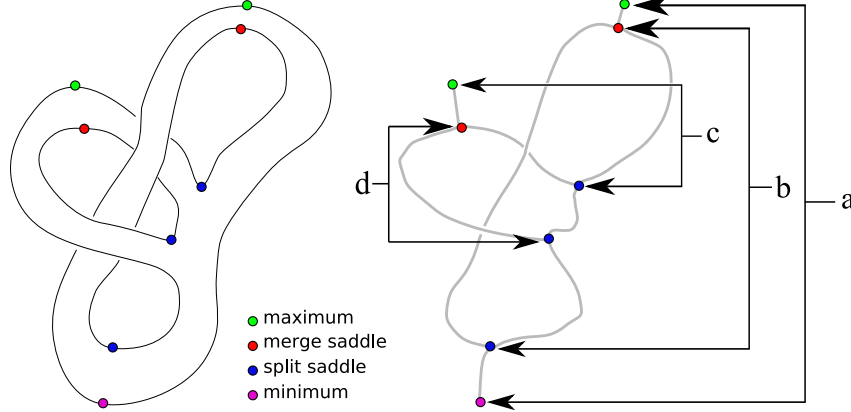


Figure 2. A 2-manifold and a corresponding Reeb graph with critical point pairs labeled.

of those simplices in the star of v that do not contain v . The lower link, $Lk_-(v)$, are those vertices $v_i \in Lk(v)$ such that $f(v_i) < f(v)$ and the edges $\overline{v_i v_j}$ such that $f(v_i) < f(v)$ and $f(v_j) < f(v)$. The upper link, $Lk_+(v)$ is defined in an analogous fashion.

The criticality of a vertex with respect to f is defined in terms of its link as demonstrated in figure 1. A vertex is regular if its lower link is a non-empty connected segment of the link. The lower link of a *minimum* is empty and the lower link of a *maximum* consists of the entire link. A k -fold *saddle* consists of $k + 1$ components along the link with $k \geq 1$. The value of f at a critical point is called a *critical value* and a level set of f containing a critical point is called a *critical level set*.

A *Morse function* is a function f that satisfies two constraints: (1) all critical points are non-degenerate and (2) for all critical points $p \neq q$, $f(p) \neq f(q)$. *Morse theory* provides techniques to explore the topology of a manifold M via functions defined on the manifold [25, 24].

The *Euler characteristic* χ relates the genus γ of M to the critical points of f :

$$\chi = \text{minima} - \text{saddles} + \text{maxima} = 2 - 2\gamma$$

If f has a single minimum and a single maximum, then it has 2γ saddles, or 2 saddles per tunnel. The level sets of f undergo topological changes at the critical points of f . Level set components originate at minima and end at maxima. Saddles are classified as *split saddles* (where a single level set component splits into two components) and *merge saddles* (where two level set components merge into one). Split saddles and minima are referred to as *parent* critical points and merge saddles and maxima are *children*.

The *Reeb graph* encodes the topology of the manifold. It is the skeleton that remains when the components of the level sets of f on M are contracted to points. It consists of a series of *nodes* connected by *arcs*. The nodes correspond to

critical points in the mesh, and the arcs correspond to mesh components. A sequence of arcs between a parent and child critical point is called a *component path*.

Critical points are often ranked in importance by *persistence*, defined as the difference in function value between parent and child critical point pairs. Traditional persistence [10] pairs split saddles with maxima and merge saddles with minima. As seen in figure 2 extended persistence [1] pairs the global maximum with the global minimum in addition to the split and merge saddles associated with each tunnel of the mesh. Note that simple component path traversal does not guarantee a successful pairing of all critical points. The mesh in figure 2 has “interleaved” tunnels where simple component path traversal of the Reeb graph beginning at two different children terminates at the same parent.

The *Jacobi set* is the set of simultaneous critical points of up to k functions on a k -manifold. Given two functions f and g defined on a 2-manifold, the Jacobi set is the set of all critical points of the restriction of f to the level sets of g . These critical points occur when the gradients of f and g are linearly dependent: $\nabla f + \lambda \nabla g = 0$. In the piecewise linear (PL) setting, functional extrema lie on mesh vertices and the Jacobi set is comprised of mesh edges.

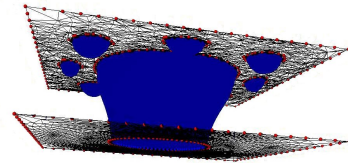


Figure 3. A 2-dimensional variational implicit function.

Variational Implicit Functions. A *radial basis function* is a real-valued function whose value depends only on the distance from a specified origin. In this paper we use the

radial basis function: $\rho(\mathbf{x}) = |\mathbf{x}|$.

A $k + 1$ -dimensional *variational implicit function* ψ [32] is used to describe the evolution of k -dimensional objects over time. It does this by minimizing a given energy while satisfying a set of input constraints: $\psi(\mathbf{c}_i) = h_i$. The input constraints consist of two types: *boundary* (where ψ is zero) and *normal* (where ψ is positive or negative). The function ψ can be expressed in terms of ρ :

$$\psi(\mathbf{x}) = \sum_{j=1}^n d_j \rho(\mathbf{x} - \mathbf{c}_j) + P(\mathbf{x})$$

Here \mathbf{c}_j are constraint locations, d_j are weights and $P(x)$ is a degree one polynomial. The set of d_j and $P(\mathbf{x})$ that satisfy the interpolation constraints can be expressed as a linear system and the weights can be computed by solving the system using the constraints as input:

$$h_i = \sum_{j=1}^n d_j \rho(\mathbf{c}_i - \mathbf{c}_j) + P(\mathbf{c}_i)$$

4 Formalization of the problem

In this section we introduce a formal framework to analyze the correctness of a parameterization method.

Definition 4.1. Let M be a 2-manifold and consider a pair of Morse functions (f, g) defined on M . We call (f, g) a *Morse field* if the following two conditions hold:

1. The non-critical level sets of f and g are 1-manifolds.
2. The restriction of f to the level sets of g and the restriction of g to the level sets of f are Morse functions.

The definition of a Morse field applies to the PL setting where Σ is a 2-dimensional simplicial complex triangulating M and f, g are piecewise linear Morse functions. The non-critical level sets of each function are piecewise linear 1-manifolds and the restrictions of f and g to each others' level sets are piecewise linearly Morse functions.

Theorem 4.2. *Iff (f, g) is a Morse field, then $\Phi = (f, g) : M \rightarrow D$ is a bijective map from Σ to D , a topologically equivalent base domain.*

Proof. If f, g form a Morse field, then Φ is a bijective map from Σ to D : f and g are finite and continuous, and cover D , therefore $\forall p' \in D$, there exists $p \in \Sigma$ such that $\Phi(p) = p'$. Assume by contradiction that $\exists p_1, p_2 \in \Sigma$ such that $p_1 \neq p_2$ and $\Phi(p_1) = \Phi(p_2)$. This implies that $f(p_1) = f(p_2)$ and $g(p_1) = g(p_2)$. Without loss of generality, assume that the function f is restricted to a level set of the function g . By definition, along this level set $g(p_1) = g(p_2)$ and f is Morse. Since $p_1 \neq p_2$ and f is

Morse then $f(p_1) \neq f(p_2)$. This is a contradiction, and therefore Φ is injective.

If Φ is a bijective map from Σ to D , then f, g form a Morse field: Assume by contradiction that Φ is a bijective map from Σ to D , and without loss of generality that the restriction of f to a level set of g is not Morse. By definition, along this level set $g(p_1) = g(p_2)$. Since the restriction of f to the level set of g is not Morse, there can exist p_1, p_2 on this level set such that $f(p_1) = f(p_2)$. This would imply that for $p_1 \neq p_2$, $\Phi(p_1) = \Phi(p_2)$, however this is a contradiction as by definition Φ is a bijective map from Σ to D . \square

Theorem 4.2 provides a formal characterization of the properties that f and g must satisfy to guarantee a bijective mapping. It is a useful exercise to consider the one-dimensional case to clarify the result. A Morse function, f , on a 1-manifold M with boundary has a maximum and a minimum that lie on the boundary and is strictly monotonic between these two points. When M is closed and without boundary, the maximum and minimum of f partition M into two regions, along which f is strictly monotonic. When M is without boundary a front/back bit is necessary to disambiguate between the partitions defined by the critical points.

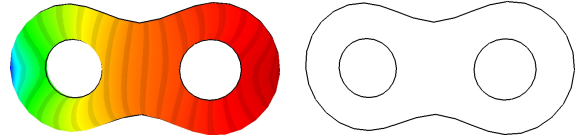


Figure 4. A minimal Jacobi set partitions M into two topologically equivalent regions.

Assuming the functions f and g form a Morse field, the 2-dimensional case is a natural extension of the 1-dimensional case. In a Morse field the generic level sets of f and g are 1-manifolds along which the restriction of the other function is Morse. Just as the critical points of a single function partition a closed 1-manifold in 1-dimension, the Jacobi set of a Morse field defined on a closed surface mesh M partitions M into two topologically equivalent regions. We call the Jacobi set of a Morse field a *minimal Jacobi set*. When M is of genus γ a minimal Jacobi set consists of $\gamma + 1$ closed loops, see figure 4. There is one loop for each tunnel in M and one loop along the perimeter or silhouette of M . In a mesh of genus $\gamma > 0$ each of these loops is non-trivial, meaning it cannot be contracted to a point.

5 Robust Cross Parameterization

Algorithm Overview. Functions that form a Morse field may be negatively impacted by the geometry of M , result-

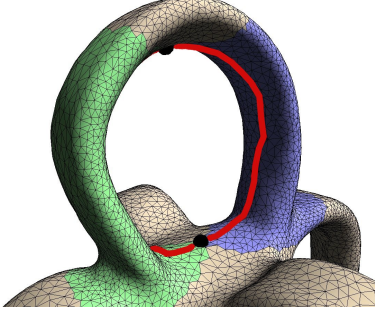


Figure 5. A Jacobi loop (shown in red) is comprised of paths traced between the associated parent and child saddles in each of the tunnel’s components (shown in blue and green).

ing in a parameterization of poor quality. We therefore consider the minimal Jacobi set to be a guideline. We compute γ non-trivial loops associated with the tunnels of M that can be used to reduce the genus of the mesh. We call these *Jacobi loops* as each one corresponds to a loop of a minimal Jacobi set.

After Jacobi loops have been computed on both input meshes M and M' , the user designates which mesh tunnels will remain features in the map by selecting landmark tunnels in addition to traditional landmark vertices. Genus reduction on each mesh is performed by cutting along the non-landmark tunnel Jacobi loops. The resulting holes are patched with scaffold faces and a common base partitioning B of all landmark features is computed. During the construction of B , a *constrained silhouette* is generated which can be integrated into the final partitioning.

A metamesh containing attribute information for M and M' is computed by embedding both meshes in B . A base patch $P \in B$ may contain one or more Jacobi loops associated with non-landmark tunnels (belonging to either M or M'). We describe the evolution of non-landmark tunnels in each such patch parametrically using a variational implicit function ψ , guaranteeing a robust evolution of tunnels regardless of their respective geometry.

Genus Reduction. Morse theory provides the tools necessary for genus reduction and, as such, functions must be defined on both M and M' . The user provides an initial coarse alignment of the input meshes via screen space orientation. A fair Morse function [26] is generated for each mesh using the screen height minimum and maximum of the models as fixed input values. We employ Simulation of Simplicity (SoS) [11] and split all saddles with degree greater than 2 as described in [9].

The Reeb graph of each mesh is computed using the approach of [4] and parent/child critical point pairs are deter-

mined using the extended persistence algorithm of [1]. The Jacobi loop for a tunnel is computed by tracing paths between the associated parent and child saddles in both of the tunnel’s components, see figure 5. Paths are traced using a restricted form of Dijkstra’s algorithm that is constrained to simplices contained in the current component. We achieve this by maintaining for each mesh face a list of the Reeb graph components it spans.

When the triangulation is coarse in relation to the proximity of the critical points, tracing shortest paths along mesh edges will fail, see figure 6 (a). To address this we conceptually run Dijkstra’s on the dual mesh. Rather than trace paths from parent v_p to child v_c critical points, we trace paths from parent to child critical faces. A parent critical face lies in the star of v_p and contains the current component label. Child critical faces are defined in an analogous fashion. Once the shortest face path, F_P , is found between a parent and child face pair, we extract a final edge path from F_P by splitting mesh edges that are incident on two faces in F_P , see figure 6 (b). If a face is both a valid parent and child we split the face as in 6 (c). Although the size of the mesh near Jacobi loops increases, we avoid the reduction of detail that can occur in previous methods [23].

Remark 5.1. Calculating paths in this manner guarantees that regardless of the coarseness of the triangulation we are able to find a Jacobi loop that does not intersect nor coincide with any other Jacobi loop.

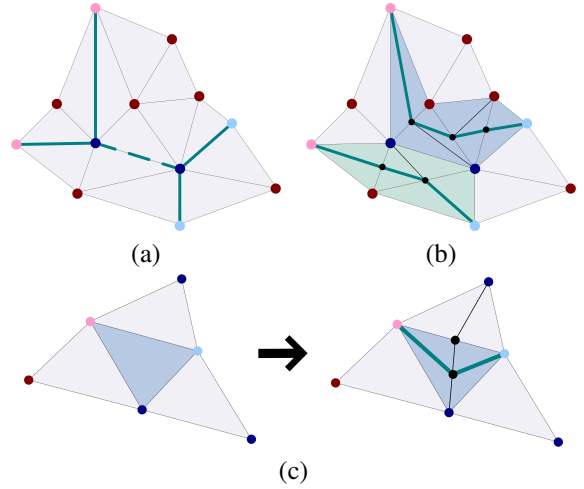


Figure 6. (a) In a coarse triangulation the possible edge paths between two different parent/child saddle pairs can intersect (the green dashed edge). (b) Face paths are computed between parent/child saddle pairs, from which a new edge path is extracted. (c) Splitting a face path of length one.

A mesh can be cut along a Jacobi loop, reducing its genus by one. The cut creates two “holes” and two Jacobi

loops per tunnel. Each hole is filled with a patch of scaffold faces that is proportional in size to the number of vertices in the Jacobi loop. The positions of interior patch vertices are assigned by solving a linear system of equations using the positions of the Jacobi loop vertices as fixed input values.

Note that Jacobi loops can belong to one of two homotopy types. Specifically, a Jacobi loop can walk “around” a tunnel or around the complementary handle, see figure 7. The homotopy type of a Jacobi loop is dependent on the function used to construct the Reeb graph and thus, control over the homotopy type of all Jacobi loops is achieved by using different functions. The user orients the mesh and is able to select a subset of the loops that result. The user can then rotate the mesh to generate different loops for the remaining tunnels. This process is repeated until γ Jacobi loops have been computed. In practice, we have found that using more than one function is seldom necessary.

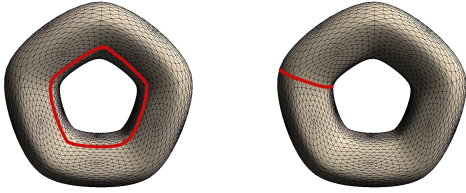


Figure 7. Jacobi loops can belong to one of two homotopy types. Shown in red, the Jacobi loop on the left walks “around” the tunnel and the Jacobi loop on the right walks around the complementary handle.

Constrained Silhouette. A minimal Jacobi set contains $\gamma + 1$ non-trivial loops. These include the γ Jacobi loops and an additional non-trivial loop along the silhouette of the model that partitions it into two topologically equivalent regions. Prior to constructing a common base partitioning B we compute an additional loop that the user can choose to integrate into B . This loop is called the *constrained silhouette* and is computed by calculating the Jacobi set of the screen space coordinate functions, x and y of the user-oriented input meshes. Using the approach of [7], we obtain a series of closed loops on the mesh, one of which contains global functional extrema. From these closed loops we construct a constrained silhouette that satisfies the following two properties:

1. It must contain the global extremal vertices of each function: x_{min} , x_{max} , y_{min} , and y_{max} .
2. It may not intersect with any Jacobi loops.

Screen space coordinate functions rarely form a Morse field and, as a result, the Jacobi set is not minimal and may violate condition (2). The loops may be jagged and

“wind” extraneously along the mesh and/or self intersect, see figure 8.

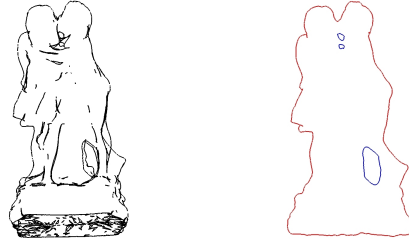


Figure 8. On the left the Jacobi set of a statue winds and self-intersects. The constrained silhouette on the right (red) does not intersect any of the Jacobi loops (blue).

To generate a constrained silhouette that satisfies both conditions, we calculate four distinct shortest paths: x_{min} to y_{max} , y_{max} to x_{max} , x_{max} to y_{min} , and y_{min} to x_{min} . We label the set of all mesh edges as E , and the set of edges belonging to a Jacobi loop are labeled E_T . All edges belonging to the Jacobi set and the neighbors of those edges are labeled E_J . To guarantee robustness against coarse triangulations we compute the constrained silhouette in a manner analogous to that which is used to compute Jacobi loops. Specifically, a restricted version of Dijkstra’s algorithm is used to trace paths on the dual mesh. We first attempt to find the shortest face path whose coincident edges belong to $E_J \setminus E_T$. When this is not possible we find the shortest face path with coincident edges belonging to $E \setminus E_T$ (this is always possible as all Jacobi loops are non-separating). The constrained silhouette consists of the four edge paths extracted from these face paths.

Base Partitioning. A common base partitioning B of all landmark vertices is computed by tracing consistent pairs of paths between feature vertices in M and M' corresponding to patch boundary edges in B . In a consistent base partitioning the topology of the path networks in M and M' are equivalent. Assuming M and M' are both of genus γ , it is observed in [30] that the path networks should contain both a minimum spanning tree of the landmark vertices and 2γ non-trivial loops prior to including any paths that form trivial loops on either mesh.

We compute the base partitioning after the user has selected landmark tunnels and landmark vertices. The non-landmark Jacobi loops are used to reduce the genus of M and M' resulting in two meshes of equal genus, α . B is computed using the α landmark Jacobi loops of M and M' as input. The mesh is cut along each of the α landmark Jacobi loops and the edges and vertices in each loop are duplicated resulting in 2α non-trivial loops on the mesh. While this operation is similar in nature to that of genus reduction,

the 2α Jacobi loops are merged at a later stage making it unnecessary to fill the resulting holes with scaffold faces.

By default two base edges are introduced into B per landmark Jacobi loop, requiring two landmark vertices per landmark tunnel. The vertices can be selected by the user or assigned automatically. We also compute a constrained silhouette which can optionally be integrated into the partitioning. If selected, this introduces four edges into B by default. However, for both the constrained silhouette and landmark Jacobi loops, the user can select different and/or additional landmark vertices, modifying the resulting number of base edges in B . When selected, the constrained silhouette partitions the mesh into two regions (front and back) and thus the remaining landmark vertices and landmark Jacobi loops are required to belong to the same region(s) in both M and M' .

To guarantee that each patch in B is a topological disk we iteratively introduce a new edge e in B until B is triangulated. For each edge $e \in B$ we embed the corresponding shortest pairs of paths from both M and M' . The edges are added in a greedy fashion based on shortest combined path length while ensuring that a minimum spanning tree of the landmark vertices is generated prior to introducing any trivial loops in either M or M' .

The paths are computed by first tracing face paths and then extracting corresponding edge paths to guarantee robustness against coarse triangulations. The paths are not allowed to intersect mesh edges embedded in base edges of B nor are they allowed to intersect non-landmark Jacobi loop edges. As each e is added to B a sweep of the mesh faces in M and M' is performed to guarantee that the landmark vertices contained in the new partitions of M and M' are equivalent. Path pairs that do not partition M and M' in an equivalent manner are disregarded. Cyclic ordering of edge paths is maintained at landmark vertices to guarantee a consistent base triangulation. Once B has been triangulated the remaining vertices of M and M' are embedded in B using traditional surface parameterization techniques.

Once a base triangulation has been computed the landmark tunnels are merged in B , M , and M' . After merging, a single edge in B may form a closed loop and/or multiple edges in B may now share the same endpoints. Edges in B that fall into either of these categories are split in two and additional edge paths are computed to complete the base triangulation.

The base partitioning introduces constraints along the edge paths embedded in edges of B . As the number of total constraints increases, the parametric distortion associated with each embedding can increase as well. In an effort to minimize this distortion we merge base triangulation faces into larger topological disks prior to embedding the input meshes. This reduces the total number of constraints, eliminating unnecessary parametric distortion without a costly

iterative step.

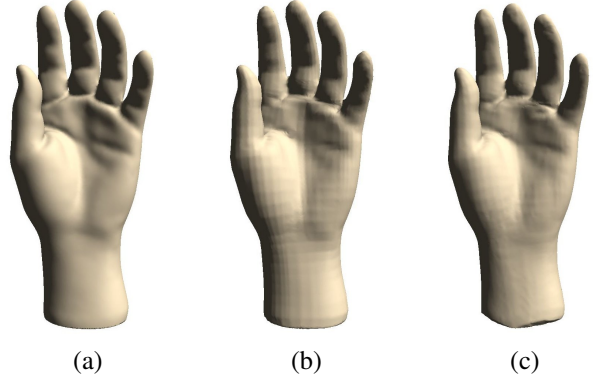


Figure 9. The full resolution hand in (a) has 38218 vertices and is genus 1. In (b) a simplified version of the hand is shown with 8808 vertices and genus 0. In (c) we show a 50% morph.

Variational Implicit Function. Topological changes in the cross parameterization are described using variational implicit functions, ψ . For each patch $P \in B$ containing at least one non-landmark Jacobi loop, ψ_P is derived automatically using the parametric descriptions of M and M' . Each ψ_P is a 3-dimensional variational function whose zero level set describes the evolution of a 2-dimensional source curve to a 2-dimensional target curve over time. We define ψ_P parametrically to guarantee that the deformation is robust regardless of mesh geometry.

The time plane corresponding to mesh M is positioned at time 0 and the time plane for M' is positioned at a distance that is equal to the parametric radius of the largest non-landmark Jacobi loop in P . This guarantees that every non-landmark tunnel has adequate time to “close” during the map. However, the position of the M' time plane can be adjusted to achieve various effects when P contains non-landmark Jacobi loops associated with both M and M' . Specifically, the proximity of the time planes (in conjunction with the parametric location of the tunnels) determines whether tunnels from M and M' will merge together or whether tunnels in M will “close” entirely prior to the “opening” of tunnels in M' . We place positive normal constraints at the base vertices of P in the time planes associated with both M and M' . This guarantees a valid ψ_P is computed regardless of differing mesh genus. Boundary constraints are positioned at all embedded vertices belonging to a non-landmark Jacobi loop and a negative normal constraint is placed at the center of the set of scaffold faces associated with each tunnel. Four additional positive constraints are positioned around each embedded non-landmark Jacobi loop (taking into consideration the position

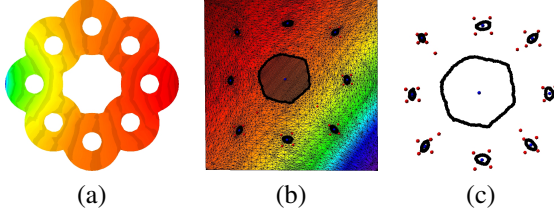


Figure 10. A 9-handle torus shown in (a) is partitioned into two patches. The parameterization of one of the patches is displayed in (b) and variational constraints for the patch are shown in (c). Positive normal constraints are drawn in red, negative normal constraints in blue and boundary constraints are drawn in black.

of the other Jacobi loops to guarantee conflict-free placement). Figure 10, shows the constraint locations for a 9-handle torus.

Metamesh. We designate as M the mesh with greater vertex count and use its connectivity as the basis for our metamesh. Attribute information for each vertex with respect to M' is calculated using the barycentric coordinates of the face in M' that contains the vertex parametrically. While this is a good starting point, there may be features in the target mesh that the source connectivity does not accurately capture. The error metric presented in [21] can be used to determine which mesh edges should be split to introduce additional detail. We have observed that edges with large edge lengths with respect to M' tend to belong to regions with the greatest amount of error. Splitting metamesh edges whose edge length with respect to M' satisfy $\|e_{M'}\| > 0.5 * \|\max(e_{M'})\|$ is a simple way to gradually refine the mesh.

6 Results

We demonstrate the results of our cross parameterization scheme by morphing between meshes of differing genus. The embedding of the surface at each stage in the morph from M to M' is computed as a simple linear interpolation of corresponding vertex positions between M and M' . The variational implicit function value of each metamesh vertex is evaluated and those vertices with negative function values are clipped out. Figure 9 (a) shows a hand data set at full resolution. During model acquisition a small tunnel was introduced between the pointer and middle fingers. A simplified version of the model that is genus 0 is shown in figure 9 (b). By not selecting the tunnel as a landmark, our algorithm produces a natural map between the two meshes. Figure 9 (c) shows the morph at 50%.

Our initial alignment scheme reduces reliance on land-

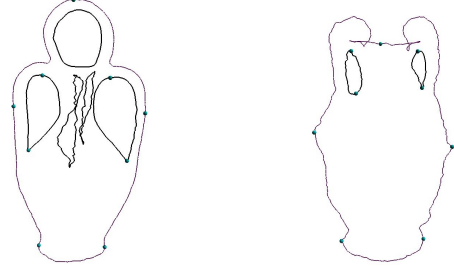


Figure 11. Nine landmark vertices are associated with the botijo and amphora meshes: 5 along the constrained silhouette and 2 per feature Jacobi loop.

marks. We illustrate this by morphing two vases of unequal genus: the botijo (genus 5) and amphora (genus 2). We select two landmark tunnels in each mesh and assign two landmark vertices per tunnel. Five landmarks are positioned along the constrained silhouette, see figure 11. The resulting morph sequence is shown in figure 12. Using the variational implicit approach, a smooth deformation is automatically calculated for the non-landmark tunnels. Figure 13 shows a series of meshes with differing genus and convoluted geometry.

7 Conclusion

In this paper we have used Morse theory to develop a formalism to characterize the correctness of a parameterization scheme. We use this formalism to develop the first provably robust method to generate maps between meshes of arbitrary and differing genus. The approach is entirely parametric and does not fail when the geometry of the input is overly convoluted. Moreover, our algorithm has no problem processing coarse meshes with high genus. To set the initial conditions, we introduce a novel alignment scheme which reduces the total number of landmarks required.

In this work our main focus has been on the formal properties and generality of the procedure that constructs a map between two meshes. In the future we plan to explore better techniques for optimizing the visual quality of the map. In particular, we plan to explore alternate path tracing methods, that consider geometric features other than path length. Furthermore, when the geometry of M and M' differs greatly, the use of linear interpolation can cause self-intersections and unnatural shape transitions. Iterative vertex optimization techniques can improve the visual quality of the map, however they are still limited when the overall difference in geometry is too great. We hope to investigate alternate interpolation schemes that could provide a more intuitive looking map.



Figure 12. A morph sequence between the botijo and the amphora meshes.

Acknowledgements. This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48. This work was supported by the Director, Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 through the Scientific Discovery through Advanced Computing (SciDAC) programs Visualization and Analytics Center for Enabling Technologies (VACET). The models used in this paper were provided by the AIM@SHAPE project.

References

- [1] P. K. Agarwal, H. Edelsbrunner, J. Harer, and Y. Wang. Extreme elevation on a 2-manifold. *Discrete Computational Geometry*, (36):553–572, 2006.
- [2] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A multi-resolution data structure for two-dimensional morse functions, May 17 2004.
- [3] C. Carner, M. Jin, X. Gu, and H. Qin. Topology-driven surface mappings with robust feature alignment. In *IEEE Visualization*, page 69, 2005.
- [4] K. Cole-McLaughlin, H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Loops in reeb graphs of 2-manifolds. In *Proceedings of the 19-th ACM Symposium on Computational Geometry (SoCG)*, pages 344–350, 2003.
- [5] D. DeCarlo and J. H. Gallier. Topological evolution of surfaces. In *Graphics Interface*, pages 194–203, 1996.
- [6] Edelsbrunner, Harer, Mascarenhas, and Pascucci. Time-varying reeb graphs for continuous space-time data. In *COMPGEO: Annual ACM Symposium on Computational Geometry*, 2004.
- [7] H. Edelsbrunner and J. Harer. Jacobi sets of multiple morse functions. In *Foundations of Computational Mathematics*, pages 37–57. Cambridge Univ. Press, 2002.
- [8] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Local and global comparison of continuous functions. In *IEEE Visualization*, pages 275–280. IEEE Computer Society, 2004.
- [9] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical morse-smale complexes for piecewise linear 2-manifolds. *Discrete Computational Geometry*, (30):87–107, 2003.
- [10] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete Computational Geometry*, (28):511–533, 2002.
- [11] H. Edelsbrunner and E. P. Mücke. Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms. In *Symposium on Computational Geometry*, pages 118–133, 1988.
- [12] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. *Advances in Multiresolution for Geometric Modelling*, pages 157–186, 2004.
- [13] I. Friedel, P. Schröder, and M. Desbrun. Unconstrained spherical parameterization. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Sketches*, page 134, New York, NY, USA, 2005. ACM Press.
- [14] C. Gotsman, X. Gu, and A. Sheffer. Fundamentals of spherical parametrization for 3d meshes. In *ACM Transactions on Graphics*, volume 22, pages 358–363. SIGGRAPH 2003, 2003.
- [15] X. Gu, S. J. Gortler, and H. Hoppe. Geometry images. *ACM Transactions on Graphics*, 21(3):355–361, July 2002.
- [16] X. Gu and S.-T. Yau. Global conformal surface parameterization. *ACM Symposium on Geometry Processing*, pages 127–137, 2003.
- [17] A. Gyulassy, V. Natarajan, V. Pascucci, P.-T. Bremer, and B. Hamann. Topology-based simplification for feature extraction from 3D scalar fields. In *IEEE Visualization*, page 68. IEEE Computer Society, 2005.
- [18] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3D shapes. In *SIGGRAPH*, pages 203–212, 2001.
- [19] M. Jin, F. Luo, and X. Gu. Computing surface hyperbolic structure and real projective structure. *Solid and Physics Modeling*, pages 613–622, 2006.

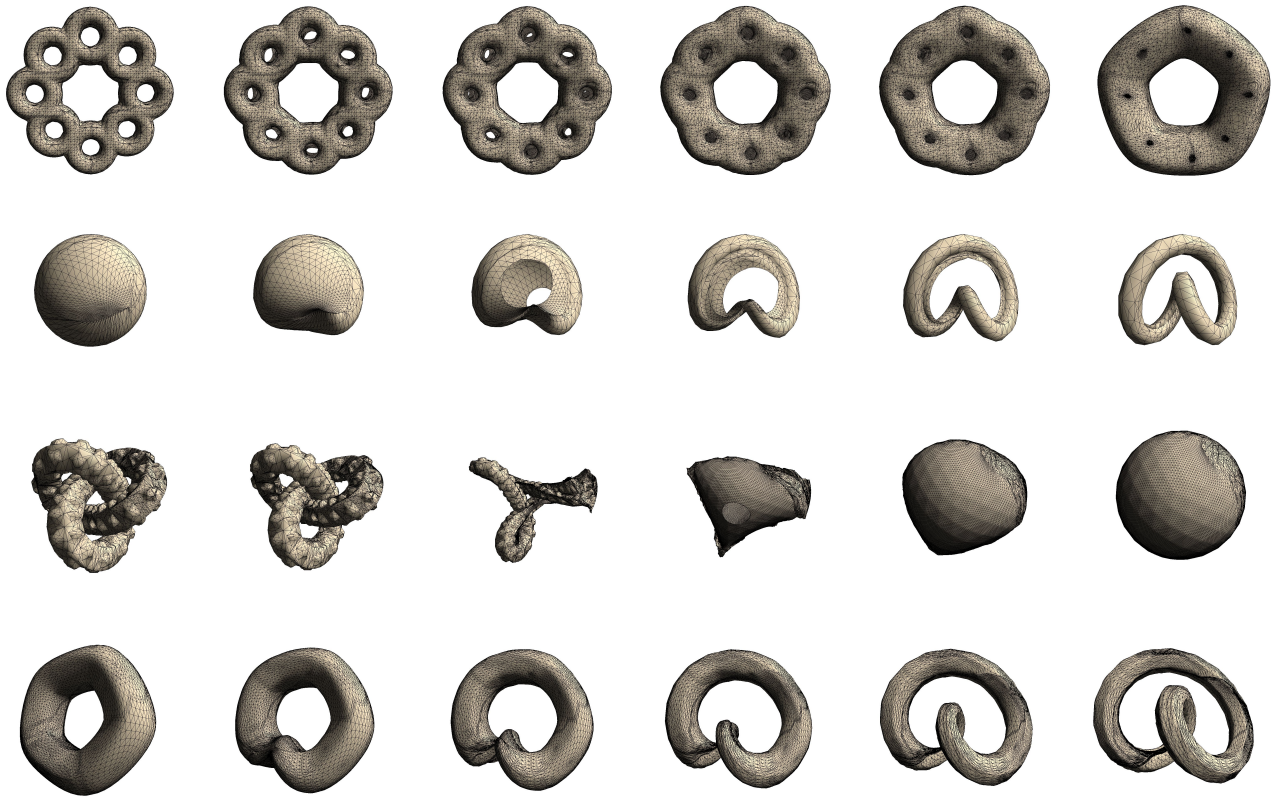


Figure 13. Regardless of convoluted geometry the algorithm is robust.

- [20] A. Khodakovsky, N. Litke, and P. Schröder. Globally smooth parameterizations with low distortion. *ACM Trans. Graph.*, 22(3):350–357, 2003.
- [21] V. Kraevoy and A. Sheffer. Cross-parameterization and compatible remeshing of 3D models. *ACM Transactions on Graphics*, 23(3):861–869, Aug. 2004.
- [22] A. Lee, D. Dobkin, W. Sweldens, and P. Schröder. Multiresolution mesh morphing. In A. Rockwood, editor, *Proceedings of the Conference on Computer Graphics (Siggraph99)*, pages 343–350, N.Y., Aug. 8–13 1999. ACM Press.
- [23] T.-Y. Lee, C.-Y. Yao, H.-K. Chu, M.-J. Tai, and C.-C. Chen. Generating genus- n -to- m mesh morphing using spherical parameterization. *Journal of Visualization and Computer Animation*, 17(3-4):433–443, 2006.
- [24] Y. Matsumoto. *An Introduction to Morse Theory*, volume 208 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, RI, 2002.
- [25] J. Milnor. *Morse Theory*, volume 51 of *Annals of mathematics studies*. Princeton University Press, Princeton, 1963.
- [26] X. Ni, M. Garland, and J. Hart. Fair morse functions for extracting the topological structure of a surface mesh. *ACM Transactions on Graphics*, 23(3):613–622, 2004.
- [27] F. Nooruddin and G. Turk. Simplification and repair of polygonal models using volumetric techniques, 1999.
- [28] G. Patane, M. Spagnuolo, and B. Falcidieno. Para-graph: Graph-based parameterization of triangle meshes with arbitrary genus. *Computer Graphics Forum*, 23(4):783–797, 2004.
- [29] E. Praun and H. Hoppe. Spherical parametrization and remeshing. *ACM Transactions on Graphics*, 22(3):340–349, July 2003.
- [30] J. Schreiner, A. Asirvatham, E. Praun, and H. Hoppe. Inter-surface mapping. *ACM Trans. Graph.*, 23(3):870–877, 2004.
- [31] A. Sheffer, C. Gotsman, and N. Dyn. Robust spherical parameterization of triangular meshes. *Computing*, 72:185–193, 2004.
- [32] G. Turk and J. F. O’Brien. Shape transformation using variational implicit functions. In *SIGGRAPH*, pages 335–342, 1999.
- [33] D. F. Wiley, N. Amenta, D. A. Alcantara, D. Ghosh, Y. J. Kil, E. Delson, W. Harcourt-Smith, K. S. John, F. J. Rohlf, and B. Hamann. Evolutionary morphing. In *IEEE Visualization*, page 55, 2005.
- [34] Z. Wood, H. Hoppe, M. Desbrun, and P. Schröder. Removing excess topology from isosurfaces. *ACM Trans. Graph.*, 23(2):190–208, 2004.
- [35] E. Zhang, K. Mischaikow, and G. Turk. Feature-based surface parametrization and texture mapping. *ACM Transactions On Graphics*, 24(1):1–27, 2005.
- [36] Q.-Y. Zhou, T. Ju, and S.-M. Hu. Topology repair of solid models using skeletons. *IEEE Transactions on Visualization and Computer Graphics*, 2007.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.