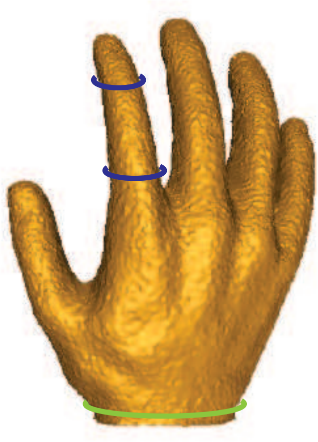
Topological Measures of Similarity

(for curves on surfaces, mostly)

Erin Wolf Chambers
Saint Louis University
erin.chambers@slu.edu

Motivation: Measuring Similarity Between Curves

How can we tell when two cycles or curves are similar to each other?



Applications

Similarity measures have many potential applications:

- Analyzing GIS data
- Map analysis and simplification
- Handwriting recognition
- Computing “good” morphings between curves
- Surface parameterizations

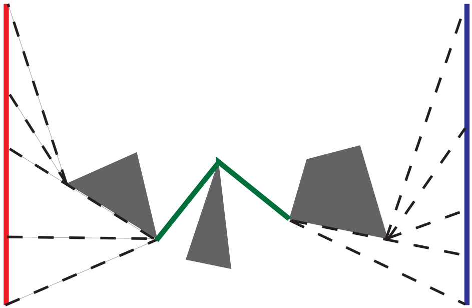
Applications

Similarity measures have many potential applications:

- Analyzing GIS data
- Map analysis and simplification
- Handwriting recognition
- Computing "good" morphings between curves
- Surface parameterizations

There are many different ways to check similarity. Most focus on either the geometry or the topology of the curve and the ambient space.

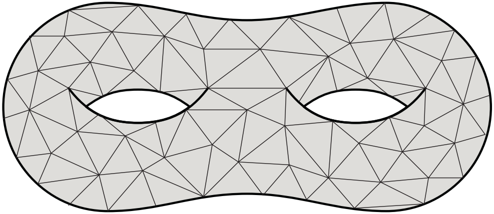
Most of this talk with focus on one of two settings. First setting:



The plane, sometimes minus a set of (polygonal) obstacles.

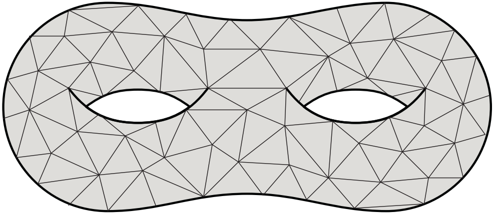
Settings

Second setting: A combinatorial or piecewise linear orientable surface.



Settings

Second setting: A combinatorial or piecewise linear orientable surface.



Any such space is homeomorphic to a sphere with a number of handles attached; we call this number the *genus* of the surface.

How we analyze algorithms

Analyzing our algorithms:

- In the plane, our algorithms will be analyzed in terms of n and m, which are the size of the input curves. If there are obstacles, we generally use p or P to denote the total complexity of the obstacles.

Analyzing our algorithms:

- In the plane, our algorithms will be analyzed in terms of n and m, which are the size of the input curves. If there are obstacles, we generally use p or P to denote the total complexity of the obstacles.
- On a surface, n will be the number of triangles in our input (which is generally an upper bound on the size of the curves, although sometimes that is separate). The value g will be the genus of the underlying surface.

Complexity of problems

We generally use big-O notation: so an algorithms which takes $3n + 1235$ operations will be called $O(n)$.

Complexity of problems

We generally use big-O notation: so an algorithms which takes $3n + 1235$ operations will be called $O(n)$.

In addition, we may use the complexity class NP. This means that any problem in this class has a polynomial time way to check if a solution is correct.

Complexity of problems

We generally use big-O notation: so an algorithms which takes $3n + 1235$ operations will be called $O(n)$.

In addition, we may use the complexity class NP. This means that any problem in this class has a polynomial time way to check if a solution is correct.

If a problem is NP-Complete, we do not know of any polynomial time algorithm; in a sense, the best solution to these problems is to try all possible solutions.

Hausdorff distance

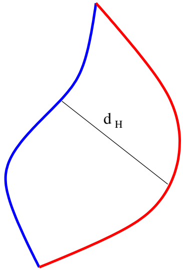
Hausdorff distance is one way to measure distance: consider any point on one curve and find the closest point on the other curve. Take the largest of all these distances.

Hausdorff distance

Hausdorff distance is one way to measure distance: consider any point on one curve and find the closest point on the other curve. Take the largest of all these distances.

More formally, given two curves $\gamma_1, \gamma_2 : [0,1] \to M$:

$$d_H(\gamma_1, \gamma_2) = \max\{\sup_{s\in[0,1]} \inf_{t\in[0,1]} d(\gamma_1(s), \gamma_2(t)), \sup_{t\in[0,1]} \inf_{s\in[0,1]} d(\gamma_1(s), \gamma_2(t))\}$$

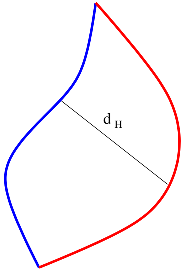


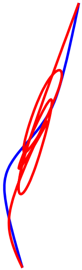
Hausdorff distance

Hausdorff distance is one way to measure distance: consider any point on one curve and find the closest point on the other curve. Take the largest of all these distances.

More formally, given two curves $\gamma_1, \gamma_2 : [0,1] \to M$:

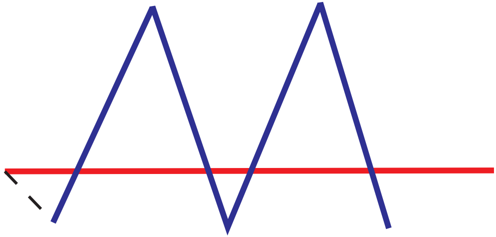
$$d_H(\gamma_1, \gamma_2) = \max\{\sup_{s\in[0,1]} \inf_{t\in[0,1]} d(\gamma_1(s), \gamma_2(t)), \\ \sup_{t\in[0,1]} \inf_{s\in[0,1]} d(\gamma_1(s), \gamma_2(t))\}$$





Fréchet Distance

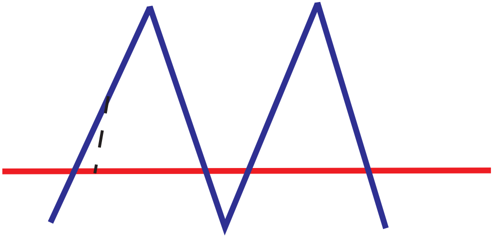
With Fréchet distance (or dog leash distance), the flow of the curve is accounted for.



Imagine a man walking along one curve and a dog along the other, with a leash always connecting them, and minimize the length of the longest leash.

Fréchet Distance

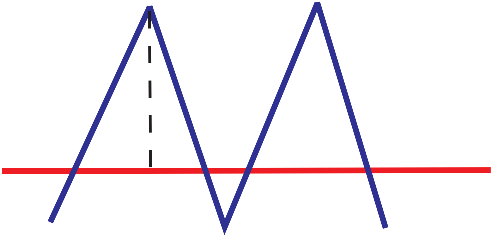
With Fréchet distance (or dog leash distance), the flow of the curve is accounted for.



Imagine a man walking along one curve and a dog along the other, with a leash always connecting them, and minimize the length of the longest leash.

Fréchet Distance

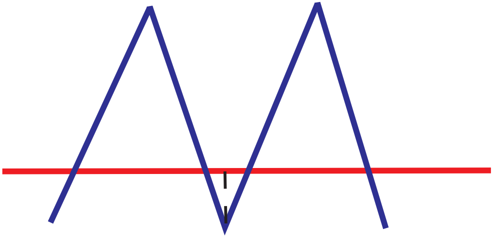
With Fréchet distance (or dog leash distance), the flow of the curve is accounted for.



Imagine a man walking along one curve and a dog along the other, with a leash always connecting them, and minimize the length of the longest leash.

Fréchet Distance

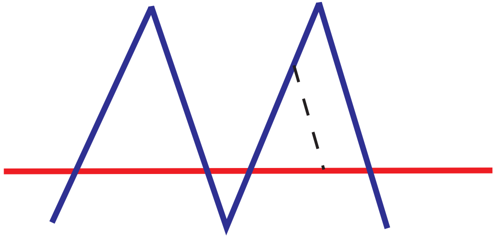
With Fréchet distance (or dog leash distance), the flow of the curve is accounted for.



Imagine a man walking along one curve and a dog along the other, with a leash always connecting them, and minimize the length of the longest leash.

Fréchet Distance

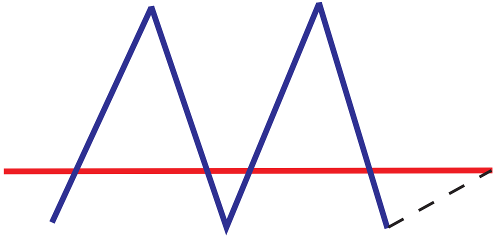
With Fréchet distance (or dog leash distance), the flow of the curve is accounted for.



Imagine a man walking along one curve and a dog along the other, with a leash always connecting them, and minimize the length of the longest leash.

Fréchet Distance

With Fréchet distance (or dog leash distance), the flow of the curve is accounted for.

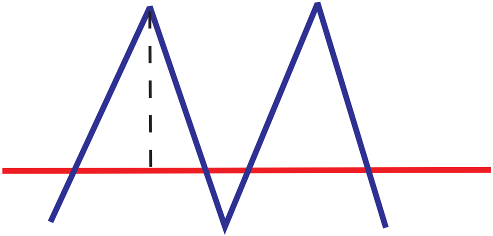


Imagine a man walking along one curve and a dog along the other, with a leash always connecting them, and minimize the length of the longest leash.

More formally, given two curves γ_1 and γ_2, the Fréchet distance is:

$$F(A,B) = \inf_{\alpha,\beta} \max_{t \in [0,1]} \{ d(\gamma_1(\alpha(t)), \gamma_2(\beta(t))) \}$$

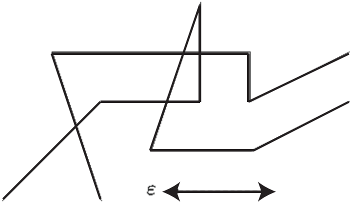
where α and β are reparameterizations of $[0,1]$.

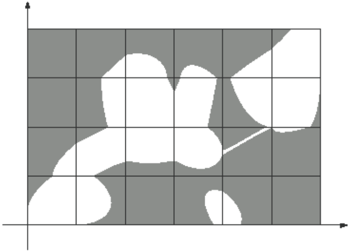


Alt and Godau gave the first algorithm to compute this for piecewise linear curves in the plane; their algorithm runs in $O(mn \log(mn))$ time.

Main tool: Free space diagram

Consider each pair of segments from the two curves, and calculate which portions are within ϵ of each other.





We build the *free space diagram* by forming the *n* by *m* grid, and determine if there is a matching that keeps the leash $\leq \epsilon$ by searching in this grid.

Fréchet distance continued

Since the initial algorithm, it has been studied extensively: applications, approximations, improved algorithms for restricted classes of curves, and lower bounds are just a few of the many results.

Fréchet distance continued

Since the initial algorithm, it has been studied extensively: applications, approximations, improved algorithms for restricted classes of curves, and lower bounds are just a few of the many results.

In addition, Fréchet distance has also been considered in higher dimensions:

- It is NP-Hard to compute the Fréchet distance between two surfaces [Godau 1998], even for polygons with holes [Buchin-Buchin-Schulz 2010].
- Still NP hard even for surfaces traced by curves [Buchin-Ophelders-Speckmann 2015].
- Finally, it is computable to compute the Fréchet distance between surfaces [Neumann 2017].

Geodesic Fréchet Distance

When not in $\mathbb{R}^d$, distance calculations are more complex since shortest paths are not unique.

Geodesic Fréchet Distance

When not in $\mathbb{R}^d$, distance calculations are more complex since shortest paths are not unique.

Definition

A geodesic is a path that avoids any obstacles and cannot be locally shortened by perturbations.

Geodesic Fréchet Distance

When not in $\mathbb{R}^d$, distance calculations are more complex since shortest paths are not unique.

Definition

A geodesic is a path that avoids any obstacles and cannot be locally shortened by perturbations.

In geodesic Fréchet distance, the leash is required to be a geodesic in the ambient space.

Geodesic Fréchet Distance

When not in $\mathbb{R}^d$, distance calculations are more complex since shortest paths are not unique.

Definition

A geodesic is a path that avoids any obstacles and cannot be locally shortened by perturbations.

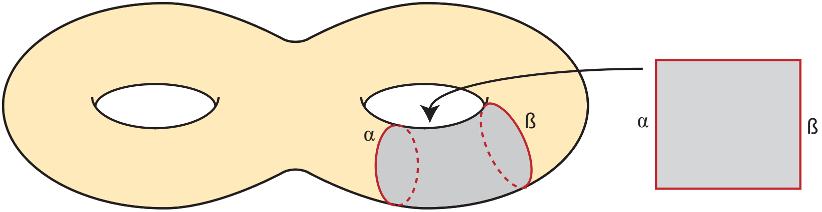
In geodesic Fréchet distance, the leash is required to be a geodesic in the ambient space.

Algorithms are known in some limited settings, such as convex polytopes [Maheshwari and Yi 2005] and simple polygons [Cook Wenk 2008]. However, much remains open.

Homotopy

Definition

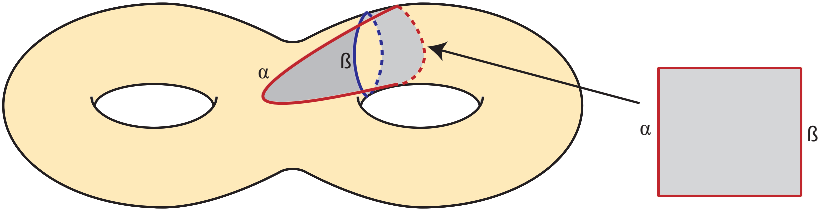
A homotopy is a continuous deformation of one path to another. More formally, a homotopy between two curves α and β on a surface M is a continuous function $H : [0,1] \times [0,1] \to M$ such that $H(\cdot, 0) = \alpha(\cdot)$ and $H(\cdot, 1) = \beta(\cdot)$.



Homotopy

Definition

A homotopy is a continuous deformation of one path to another. More formally, a homotopy between two curves α and β on a surface M is a continuous function $H : [0,1] \times [0,1] \to M$ such that $H(\cdot, 0) = \alpha(\cdot)$ and $H(\cdot, 1) = \beta(\cdot)$.



Testing if two curves are homotopic

Testing if two curves are homotopic has been studied in both of our settings.

- Cabello et al (2004) give an algorithm to test if two paths in the plane minus a set of obstacles are homotopic in $O(n^{3/2} \log n)$ time.
- Given a graph cellularly embedded on a surface and two closed walks on that graph, there is an $O(n)$ time algorithm to decide if the two walks are homotopic [Dey and Guha 1999, Lazarus and Rivaud 2011, Erickson and Whittlesey 2012].

Combinatorially optimal homotopies

There is work [Chang-Erickson 2016] on finding the "best" homotopy, as well; usually, this involves minimizing number of simplifications moves to untangle a curve.

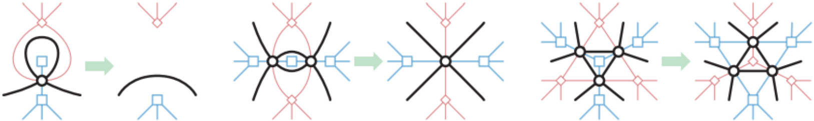
Figure 1.1. Homotopy moves 1→0, 2→0, and 3→3.

Combinatorially optimal homotopies

There is work [Chang-Erickson 2016] on finding the "best" homotopy, as well; usually, this involves minimizing number of simplifications moves to untangle a curve.

Figure 1.1. Homotopy moves 1→0, 2→0, and 3→3.

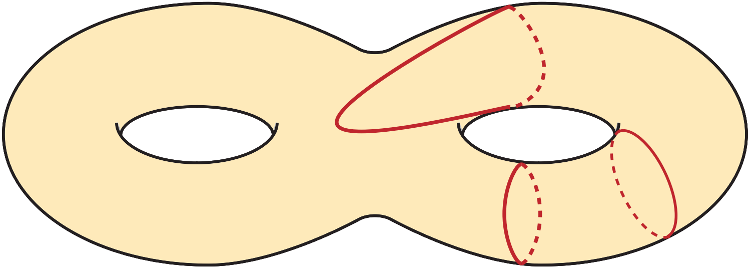
In the plane, they prove this is $\Theta(n^{3/2})$.



This connects to older results [Steinitz 1916, Francis 1969, Truemper 1989, Feo and Provan 1993, Noble and Welsh 2000], and electrical moves on the medial graph of the input planar graphs.

Beyond testing homotopy

However, in many applications we'd like to include more of a notion of the geometry of the underlying space, as well.



Homotopic Fréchet Distance

The definition of Fréchet distance or geodesic Fréchet distance will directly generalize to surfaces, but does not take homotopy into account. Essentially, either definition allows the leash to jump discontinously.

Homotopic Fréchet Distance

The definition of Fréchet distance or geodesic Fréchet distance will directly generalize to surfaces, but does not take homotopy into account. Essentially, either definition allows the leash to jump discontinously.

Homotopic Fréchet is adds a constraint that the curves must be homotopic, and the leashes must move continuously in the ambient space [C.-Colin de Verdiére-Erickson-Lazard-Lazarus-Thite 2009].

Homotopic Fréchet Distance

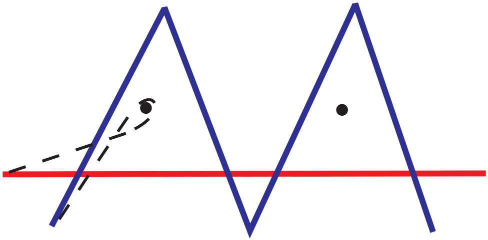
The definition of Fréchet distance or geodesic Fréchet distance will directly generalize to surfaces, but does not take homotopy into account. Essentially, either definition allows the leash to jump discontinously.

Homotopic Fréchet is adds a constraint that the curves must be homotopic, and the leashes must move continuously in the ambient space [C.-Colin de Verdiére-Erickson-Lazard-Lazarus-Thite 2009].

Intuitively, curves with small homotopic Fréchet distance will be close both geometrically and topologically.

The homotopic Fréchet distance is the length of the shortest leash we can can use for our homotopy. Formally,

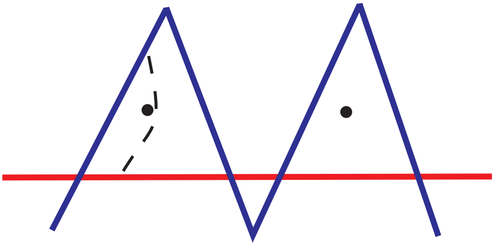
$$d_F(\gamma_1, \gamma_2) = \inf_{\text{homotopies } H} \{\sup\{|H(\cdot, t))| \mid t \in [0, 1]\}\}$$



Homotopic Fréchet Distance

The homotopic Fréchet distance is the length of the shortest leash we can can use for our homotopy. Formally,

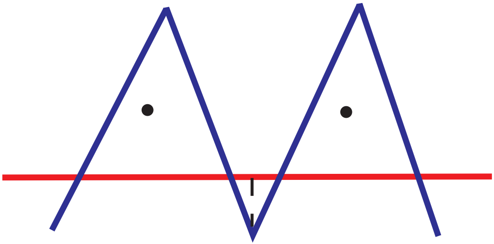
$$d_F(\gamma_1, \gamma_2) = \inf_{\text{homotopies } H} \{\sup\{|H(\cdot, t))| \mid t \in [0, 1]\}\}$$



Homotopic Fréchet Distance

The homotopic Fréchet distance is the length of the shortest leash we can can use for our homotopy. Formally,

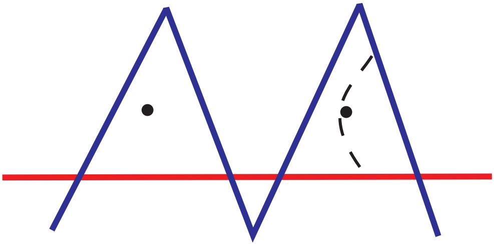
$$d_F(\gamma_1, \gamma_2) = \inf_{\text{homotopies } H} \{\sup\{|H(\cdot, t))| \mid t \in [0, 1]\}\}$$



Homotopic Fréchet Distance

The homotopic Fréchet distance is the length of the shortest leash we can can use for our homotopy. Formally,

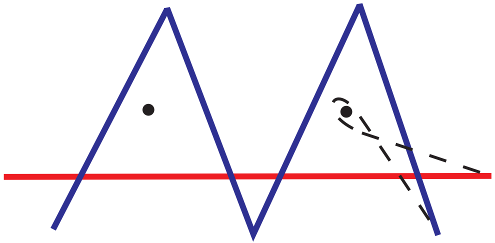
$$d_F(\gamma_1, \gamma_2) = \inf_{\text{homotopies } H} \{\sup\{|H(\cdot, t))| \mid t \in [0, 1]\}\}$$



Homotopic Fréchet Distance

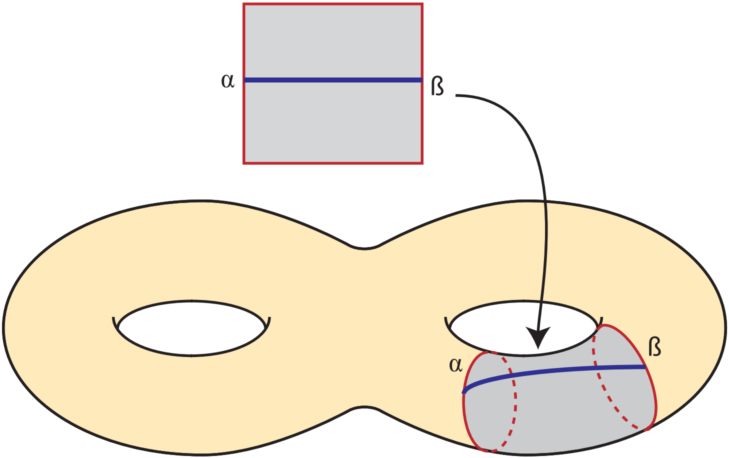
The homotopic Fréchet distance is the length of the shortest leash we can can use for our homotopy. Formally,

$$d_F(\gamma_1, \gamma_2) = \inf_{\text{homotopies } H} \{\sup\{|H(\cdot, t))| \mid t \in [0,1]\}\}$$



Homotopic Fréchet Distance on a Surface

We could just have easily called this the *width* of the homotopy:



(Note: it is not known how to compute this on surfaces at all.)

There is a polynomial time algorithm algorithm to compute the homotopic Fréchet Distance between two polygonal curves in the plane minus a set of polygonal obstacles [C.-Colin de Verdiére-Erickson-Lazard-Lazarus-Thite 2009].

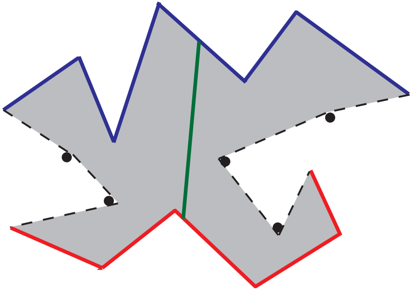
There is a polynomial time algorithm algorithm to compute the homotopic Fréchet Distance between two polygonal curves in the plane minus a set of polygonal obstacles [C.-Colin de Verdiére-Erickson-Lazard-Lazarus-Thite 2009].

The algorithm has some similarities to the work of Alt and Godau, but is considerably more complex since there are an infinite number of homotopy classes to consider.

Key lemma

Lemma

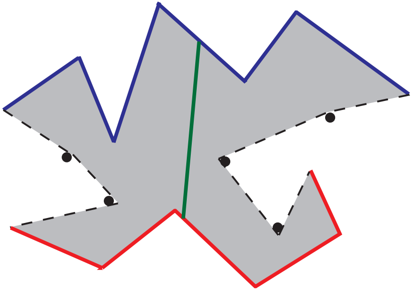
When obstacles are points, an optimal homotopy class contains a straight line segment.



Key lemma

Lemma

When obstacles are points, an optimal homotopy class contains a straight line segment.



This allows us to brute force a set of possible homotopy classes which could be optimal, by trying all straight line segments.

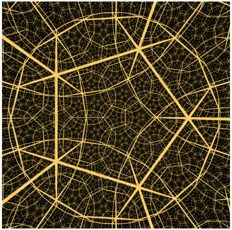
Homotopic Fréchet Distance in other settings

It is unlikely our algorithm will generalize to surfaces, since it heavily relies on nonpositive curvature.

Homotopic Fréchet Distance in other settings

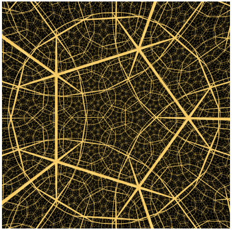
It is unlikely our algorithm will generalize to surfaces, since it heavily relies on nonpositive curvature.

We can generalize the key lemmas to any surface of nonpositive curvature. However, the algorithmic tools in those settings are lacking.



It is unlikely our algorithm will generalize to surfaces, since it heavily relies on nonpositive curvature.

We can generalize the key lemmas to any surface of nonpositive curvature. However, the algorithmic tools in those settings are lacking.

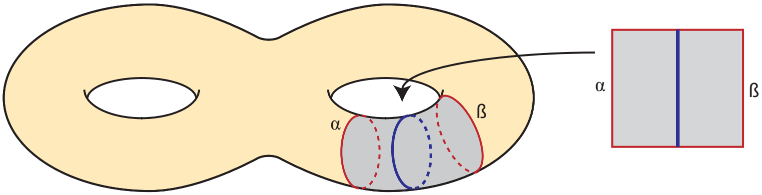


In an upcoming paper (joint with Arnaud de Mesmay and Tim Ophelders), we are able to show the problem is in NP in the planar or combinatorial surface setting when the beginning and ending leash are fixed, but this is slightly different than the planar case, where these leashes are not fixed.

Height of a homotopy

The height of a homotopy is an orthogonal definition to homotopic Fréchet distance:

$$d_{HH}(\gamma_1, \gamma_2) = \inf_{\text{homotopies } H} \{\sup\{|H(s, \cdot)| \mid s \in [0,1]\}\}$$



Computing homotopy height

No algorithm is known to compute the homotopy height between two curves in any setting.

We do know that the problem is in NP (forthcoming joint work with Gregory Chambers, Arnaud de Mesmay, Tim Ophelders, and Regina Rotman).

Computing homotopy height

No algorithm is known to compute the homotopy height between two curves in any setting.

We do know that the problem is in NP (forthcoming joint work with Gregory Chambers, Arnaud de Mesmay, Tim Ophelders, and Regina Rotman).

If you consider the setting where your curves are the boundaries of a triangulated disk, it is closely connected to parameters such as cut width which are known to be NP-Complete, but the reductions do not quite work for this problem.

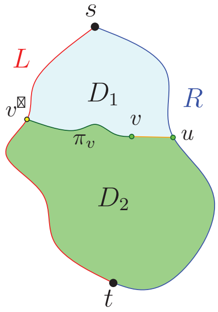
Approximation algorithms

[Har-Peled-Nayyeri-Salavatipour-Sidiropoulos 2012] give an $O(\log n)$ approximation algorithm for computing both the homotopy height and the homotopic Fréchet distance between two curves on a PL surface.

Approximation algorithms

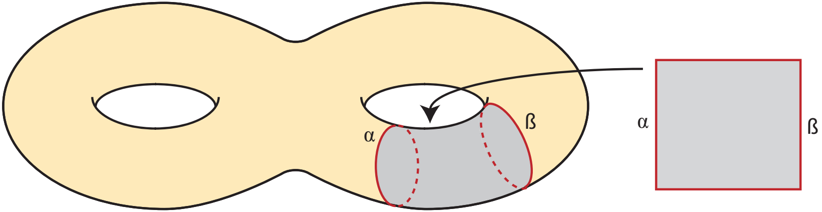
[Har-Peled-Nayyeri-Salavatipour-Sidiropoulos 2012] give an $O(\log n)$ approximation algorithm for computing both the homotopy height and the homotopic Fréchet distance between two curves on a PL surface.

They use a clever divide and conquer algorithm based on shortest paths for homotopy height, and then use this algorithm as a subroutine to solve homotopic Fréchet distance.



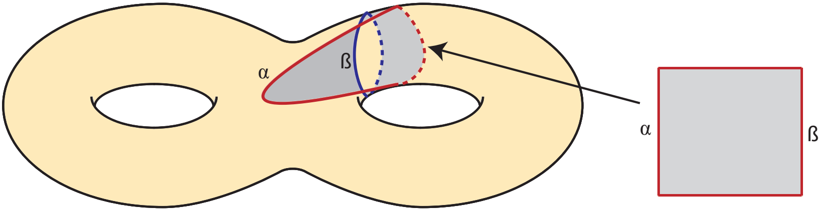
Area of a homotopy

Instead of focusing on the length or width, we can also examine the total area swept by a homotopy [C-Wang 2013].



Computing homotopy area

Surprisingly, this measure is much more tractable on surfaces than any other measure which takes topology into account, even for non-disjoint cycles.



Definition

More formally, given a homotopy H, the area of H is defined as:

$$\text{Area}(H) = \int_{s \in [0,1]} \int_{t \in [0,1]} \left| \frac{dH}{ds} \times \frac{dH}{dt} \right| dsdt$$

Definition

More formally, given a homotopy H, the area of H is defined as:

$$\text{Area}(H) = \int_{s\in[0,1]} \int_{t\in[0,1]} \left| \frac{dH}{ds} \times \frac{dH}{dt} \right| dsdt$$

We are then interested in the smallest such value: $\inf_H \text{Area}(H)$.

Definition

More formally, given a homotopy H, the area of H is defined as:

$$\text{Area}(H) = \int_{s\in[0,1]} \int_{t\in[0,1]} \left| \frac{dH}{ds} \times \frac{dH}{dt} \right| dsdt$$

We are then interested in the smallest such value: $\inf_H \text{Area}(H)$.

Note that in generally, this is an improper integral, and the value for any H is not necessarily even finite.

Douglas and Rado's work

Douglas and Rado (1930's) were the first to consider this problem, as a variant of Plateau's problem (1847) dealing with soap bubbles and minimal surfaces.

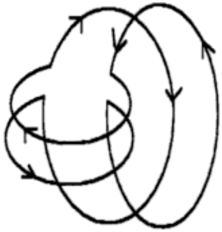


Fig. 4.1

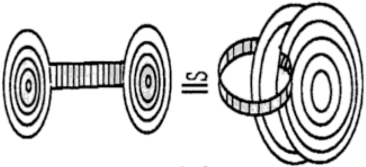


Fig. 4.2

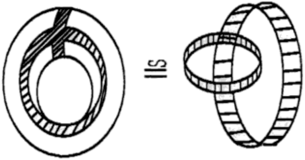
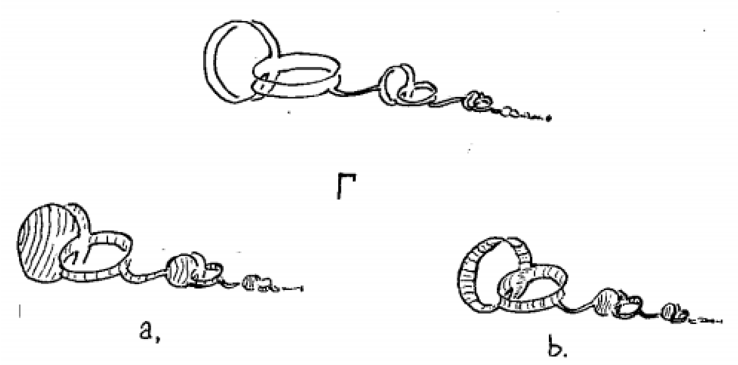


Fig. 4.3

[Minimal sub manifolds and related topics, Y. L. Xin]

There is an additional problem in that to find the infimum, we might have a pathological case where a sequence of good H's converge to something that is not even continuous.



[Lectures on Minimal Submanifolds, H. B. Lawson]

Douglas' theorem

They developed a restricted version using Dirichlet integrals (or energy integrals) which allow control over the parameterizations of the minimal surfaces. These integrals not only minimize area, but also ensure (almost) conformal parameterizations in the space.

Douglas' theorem

They developed a restricted version using Dirichlet integrals (or energy integrals) which allow control over the parameterizations of the minimal surfaces. These integrals not only minimize area, but also ensure (almost) conformal parameterizations in the space.

Theorem

Let γ be a finite Jordan curve in $\mathbb{R}^n$. Then there exists a continuous map $\Gamma : \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\} \to \mathbb{R}^n$ such that:

1. Γ *maps the boundary of the disk monontically onto γ.*
2. Γ *is harmonic and almost conformal*
3. Γ *realizes the infimum of all areas*

Douglas' theorem

They developed a restricted version using Dirichlet integrals (or energy integrals) which allow control over the parameterizations of the minimal surfaces. These integrals not only minimize area, but also ensure (almost) conformal parameterizations in the space.

Theorem

Let γ be a finite Jordan curve in $\mathbb{R}^n$. Then there exists a continuous map $\Gamma : \left\{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\right\} \to \mathbb{R}^n$ such that:

1. *Γ maps the boundary of the disk monontically onto γ.*
2. *Γ is harmonic and almost conformal*
3. *Γ realizes the infimum of all areas*

(Well, I'm hiding a few details about the Dirichlet integrals here...)

Necessary assumptions

Our setting is much simpler - we are either in $\mathbb{R}^2$ or a piecewise linear surface. However, we do need some assumptions in order for the minimum area homotopy to exist.

- We must assume that H is continuous and piecewise differentiable (so it is differentiable everywhere except at a finite set of points and arcs).

Necessary assumptions

Our setting is much simpler - we are either in $\mathbb{R}^2$ or a piecewise linear surface. However, we do need some assumptions in order for the minimum area homotopy to exist.

- We must assume that H is continuous and piecewise differentiable (so it is differentiable everywhere except at a finite set of points and arcs).
- We must also assume the homotopy is monotone along the boundary of the domain and is regular on the interior (meaning intermediate curves are "kink-free").

Necessary assumptions

Our setting is much simpler - we are either in $\mathbb{R}^2$ or a piecewise linear surface. However, we do need some assumptions in order for the minimum area homotopy to exist.

- We must assume that H is continuous and piecewise differentiable (so it is differentiable everywhere except at a finite set of points and arcs).
- We must also assume the homotopy is monotone along the boundary of the domain and is regular on the interior (meaning intermediate curves are "kink-free").
- Finally, we will assume our input curves (on M) are simple and have a finite number of piecewise analytic components. (In practice, they will simply be PL curves.)

Running time in the plane

Let I be the number of intersections and n be the complexity of the input curves.

Running time in the plane

Let I be the number of intersections and n be the complexity of the input curves.

We give an algorithm that can be implemented in $O(I^2 n)$ time using dynamic programming, which simply builds up the sets of anchor points iteratively and uses previous solutions to speed up future computation.

Running time in the plane

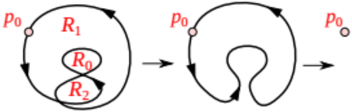
Let I be the number of intersections and n be the complexity of the input curves.

We give an algorithm that can be implemented in $O(I^2 n)$ time using dynamic programming, which simply builds up the sets of anchor points iteratively and uses previous solutions to speed up future computation.

However, this can be improved to $O(I^2 \log I)$ time with $O(I \log I + n)$ preprocessing if we are more careful about how we compute the winding numbers.

More recent algorithms for homotopy area

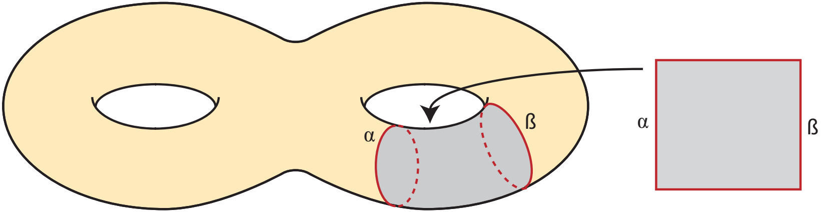
There has also been recent work to compute the best area homotopy when the input curve is not so "nice", but is an immersion of a disk into the plane.



- One result [Nie 2014] connects this problem to the weighted cancellation norm, which is a very combinatorial way to covert the best homotopy into a series of reduction moves on a word problem. The result is a polynomial time algorithm.
- Another [Fasy-Karakoc-Wenk 2016] consider a different approach which is more geometric, building up an exponential time algorithm, although perhaps faster dynamic programming techniques can speed this up.

Homotopy area on a surface

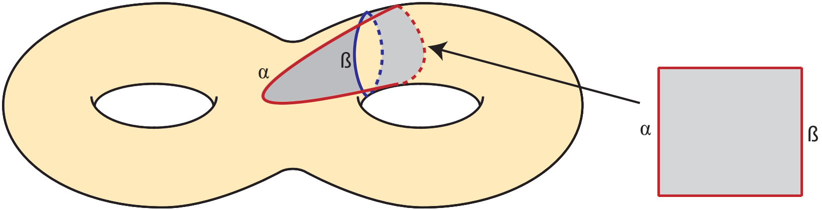
Our paper [C.-Wang] also considers the algorithm for surfaces, which builds upon the planar algorithm.



Consider two homotopic curves on a triangulated surface M with positive genus.

Homotopy area on a surface

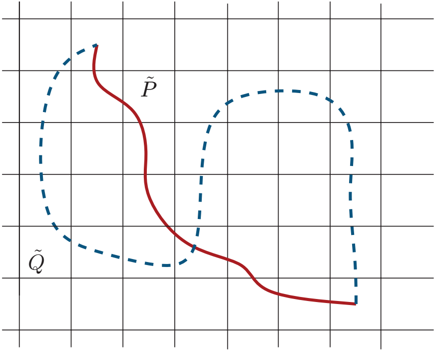
Our paper [C.-Wang] also considers the algorithm for surfaces, which builds upon the planar algorithm.



Consider two homotopic curves on a triangulated surface M with positive genus.

Lifting P and Q

If we fix a lift for the endpoints of P and Q in the universal cover $U(M)$, then $P \circ Q$ lifts to a unique closed curve in $U(M)$. Therefore, any homotopy between P and Q on M will correspond to a homotopy between their lifts in $U(M)$ with the same area.



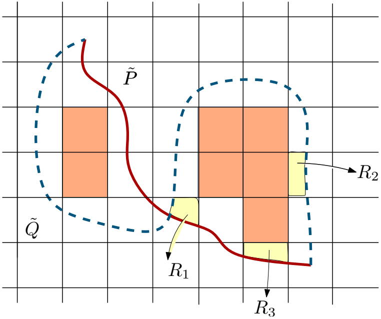
Algorithm for surfaces

We construct a portion of the universal cover which contains the lifts of P and Q as well as the regions inside their concatenation.

Algorithm for surfaces

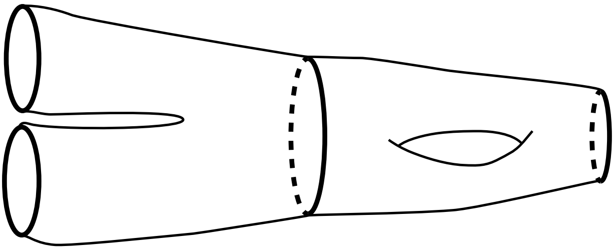
We construct a portion of the universal cover which contains the lifts of P and Q as well as the regions inside their concatenation.

We then use our planar algorithm in $U(M)$, since similar results about the winding number will hold. Since we can simplify much of the interior of the regions in our representation, the total running time here is $O(gK \log K + I^2 \log I + In)$.



Using homology?

- Homology is a coarser invariant than homotopy - all homotopies produce homologies, but not all homologies come from homotopies.
- In general, much more tractable - reduces to a linear algebra problem, and software is widely available and highly optimized.
- Potentially much wider applications: works for cobordisms, arbitrary dimension submanifolds of arbitrary dimension manifolds, etc.



How to compute homology area

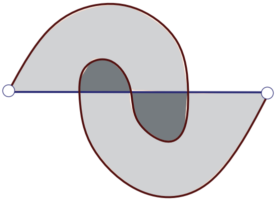
Formally (joint work with Mikael Vejdemo Johansson, and also considered in more limited settings in Dey, Hirani and Krishnamoorthy):

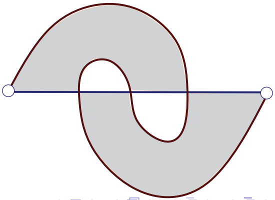
- Given cycles α and β, try to compute z such that $dz = \alpha - \beta$.
- Goal: compute z with a smallest area. Recall that d is a linear operator, and z and $\alpha - \beta$ are vectors.
- Optimization problem is then:
  arg min_z (area z), subject to $dz = \alpha - \beta$.

How to compute homology area

Formally (joint work with Mikael Vejdemo Johansson, and also considered in more limited settings in Dey, Hirani and Krishnamoorthy):

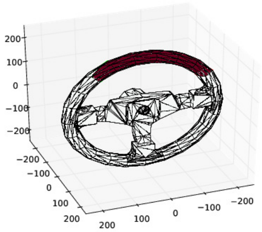
- Given cycles α and β, try to compute z such that $dz = \alpha - \beta$.
- Goal: compute z with a smallest area. Recall that d is a linear operator, and z and $\alpha - \beta$ are vectors.
- Optimization problem is then:
  arg min_z (area z), subject to $dz = \alpha - \beta$.
- Note again that this is NOT the same as homotopy area, at least for $d \leq 3$:

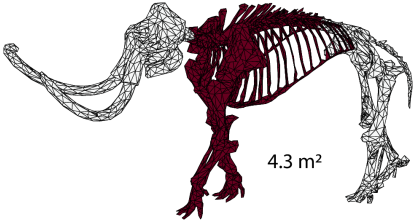




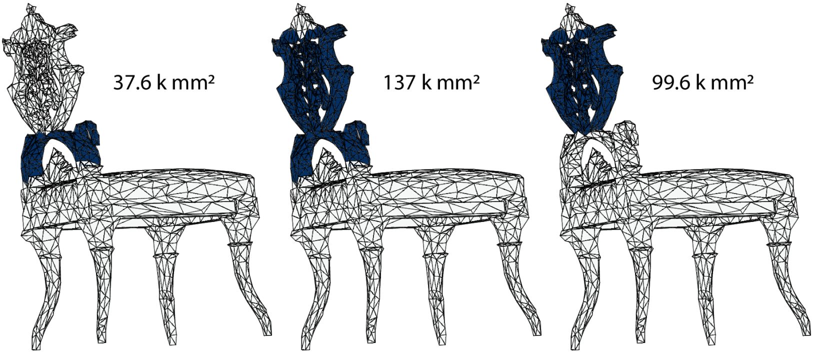
Final algorithm for homology area

In matrix multiply time, we can compute the best area homology on meshes:

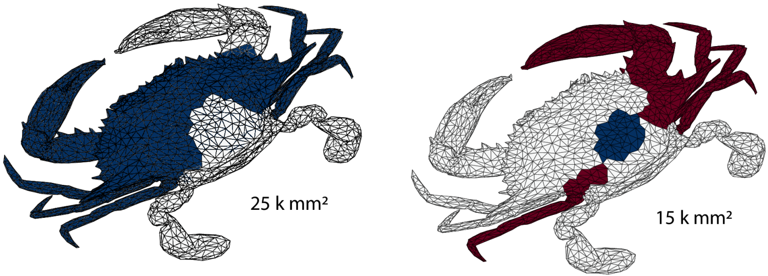




Chair model



Crab model



Isotopy

Definition

An *isotopy* is a homotopy H such that for each fixed time t, $H(x, t)$ is a homeomorphism.

Isotopy

Definition

An *isotopy* is a homotopy H such that for each fixed time t, $H(x, t)$ is a homeomorphism.

A *homeomorphism* is a function which is a continuous bijection where the inverse is also continuous. In our setting, this will mean that every intermediate curve in the homotopy must also have an image that is simple.

Testing Isotopy

Algorithmic results here are much newer.

Testing Isotopy

Algorithmic results here are much newer.

[Colin de Verdière-de Mesmay 2012] give the first algorithms to test if two graphs are isotopic.

Testing Isotopy

Algorithmic results here are much newer.

[Colin de Verdière-de Mesmay 2012] give the first algorithms to test if two graphs are isotopic.

- In the plane minus polygonal obstacles, the test takes $O(n^{3/2} \log n)$ time.
- On a combinatorial surface, the test takes $O(n)$ time.

Testing Isotopy

Algorithmic results here are much newer.

[Colin de Verdière-de Mesmay 2012] give the first algorithms to test if two graphs are isotopic.

- In the plane minus polygonal obstacles, the test takes $O(n^{3/2} \log n)$ time.
- On a combinatorial surface, the test takes $O(n)$ time.

Note that the isotopy is fixed in the sense that you must indicate which vertices map to each other under the isotopy.

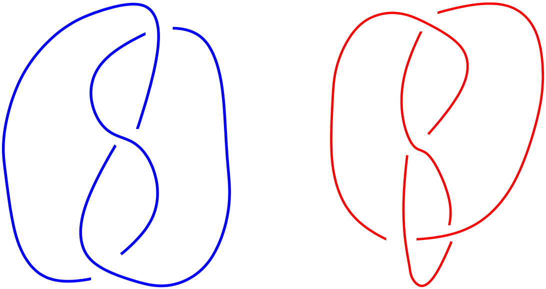
Isotopic Fréchet Distance

[C.-Ju-Letscher 2009] introduced the idea of isotopic Fréchet distance:

$$\mathcal{I}(A,B) = \inf_{\substack{h : M \times I \to M \\ h(\cdot,t) \text{ homeomorphism} \\ h(x,0)=x \ \forall x \in X \\ h(A,1)=B}} \max_{x \in X} \operatorname{len} h(x,\cdot)$$

In other words, what's the longest trajectory in an ambient isotopy?

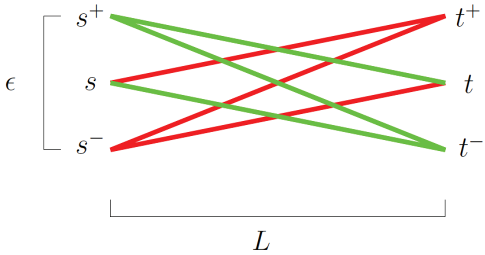
If A and B are not ambiently isotopic then $\mathcal{I}(A, B) = \infty$.



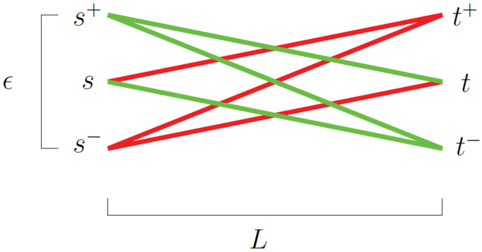
Homotopic versus Isotopic Fréchet Distance

Proposition For any $L > 0$ and $\epsilon \in (0, L/2)$ there exists a pair of curves $C_1, C_2 \in \mathbb{R}^2$ with

$$\begin{aligned} \mathcal{F}(C_1, C_2) = \mathcal{H}(C_1, C_2) &= \epsilon \\ \mathcal{I}(C_1, C_2) &\geq \frac{2}{9} L \end{aligned}$$

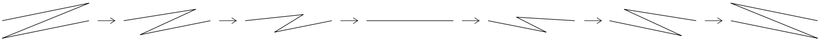


The best homotopy versus an isotopy



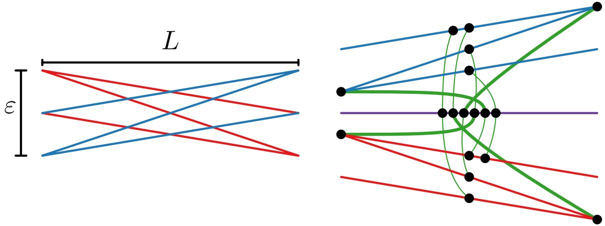
Homotopy:

Isotopy?:



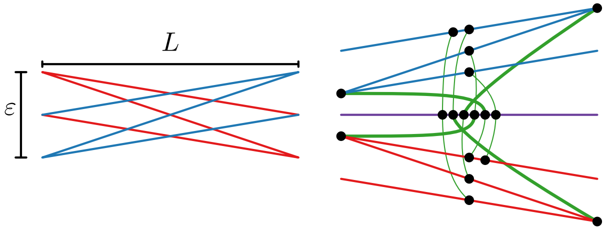
The best homotopy versus an isotopy

Actually, the best isotopy is even more complicated! The prior picture gave a distance of $\sqrt{L^2 + \epsilon^2}$. This was off by a factor of roughly 2 [Buchin-C.-Ophelders-Speckmann 2017]:



The best homotopy versus an isotopy

Actually, the best isotopy is even more complicated! The prior picture gave a distance of $\sqrt{L^2 + \epsilon^2}$. This was off by a factor of roughly 2 [Buchin-C.-Ophelders-Speckmann 2017]:



In this work, we also consider restricted versions of the problem, and compute optimal isotopies if there is a direction in which both input curves are monotone.

Other notions of morphing

Other types of measures may be worth considering, as topological notions are not often incorporated in this literature:

- Geodesic width [Efrat, Guibas, Har-Peled, Mitchell, and Murali 2002] is a notion of deformation where intermediate curves may not cross the initial input curves, and the morph must stay within the area enclosed by the initial and final leash (combined with the curves). Since these are geodesic, again the leashes won't cross. However, the two input curves are also not allowed to cross each other.
- There are many algorithms (i.e. [Angelini et al, 2014] that seek to compute a morph which bounds the number of steps in the morph; these don't really consider the geometry as much, but perhaps could use tools or be connected to more combinatorial notions of homotopy.

Open questions (part 1)

- Other than homology area, very few of these algorithms have been implemented, despite many practical applications.
- There is no algorithm to compute homotopic Fréchet distance on surfaces (or even polyhedra).
- Height of a homotopy algorithms are also open; all that is known is an O(log n) approximation and that it's in NP.
- (Perhaps both are even NP-Complete...)
- It is unknown how to compute homotopy area between cycles on surfaces.

Open questions (continued)

- Testing isotopy is understood, but using it as a measure of similarity is pretty wide open.
- All of these could be used for finding median trajectories or perhaps clustering - recent work uses homology area (practical) and homotopy area (not so practical), but applications areas could motivate new directions.
- Can any of these be made tractable for curves on 3-manifolds, or even 3-manifolds which are embedded in $\mathbb{R}^3$? (Possibly instead of area we may need a more combinatorial notion, such as Hsien's result - a "best" homotopy might be one with fewer uncrossing moves, then.)