21 Homotopy Testing on Surface Maps

Let’s return to one of the earliest problems we saw this semester: Given two curves in the same surface, decide whether they are homotopic, meaning one can be continuously deformed into the other. Here I’ll describe a linear-time algorithm that slightly improves a classical algorithm of Max Dehn (1911).

To keep things simple, I’ll focus on the following special case: Given a closed curve γ in some surface S, is γ contractible in S? That is, can γ be continuously deformed on the surface S to a single point?

This question is obviously trivial in the plane or the sphere, and it turns out to be easy on the projective plane, torus, or Klein bottle, so without loss of generality, I will the underlying surface has negative Euler characteristic χ < 0. Let ¯g = 2 − χ denote the Euler genus of the input surface. The Euler genus is equal to the standard genus if the surface is non-orientable, and twice the standard genus if the surface is orientable.

21.1 What’s a “curve on a surface”?

But before we can even start talking about algorithms, we have to nail down the phrase “given two curves” and “given a surface”. Our earlier planar homotopy algorithm assumed curves were given as polygons, specified as a sequence of vertex coordinates; while it is possible to impose coordinates on surfaces that would permit a natural generalization of “polygons”, imposing geometry is almost always both wasteful and unnecessary.1

It is usually simpler and more efficient to treat curves on surfaces purely combinatorially, representing surfaces as maps (either rotation systems or reflection systems), and curves using one of two natural combinatorial models:

- **Traversal**: We consider only curves that are walks in the graph of Σ; curves cannot intersect faces, and if a curve intersects an edge, it must traverse that edge monotonically from one end to the other. We can represent any such curve as an alternating sequence of vertices and darts (directed edges). If the edges of Σ are weighted, the length of a curve α is the sum of the weights of the edges that α traverses, counted with appropriate multiplicity.

- **Crossing**: We consider only curves that intersect the edges of Σ transversely, away from the vertices, and whose intersection with each face of Σ is simple (injective). We can represent any such curve as an alternating sequence of faces and knives (oriented edges). If the edges of Σ are weighted, the length of a curve α is the sum of the weights of the edges that α crosses, counted with appropriate multiplicity.

These two models are clearly dual to each other; a crossing curve in Σ is equivalent to—or more formally, IS—a traversing curve in the dual map Σ∗, and vice versa. Which curve model we choose is strictly a matter of convenience. For the rest of this lecture, I’ll use the traversal model.

21.2 Spur and Face Moves

Even though we do not allow the input curves to intersect the interiors of faces, we cannot impose the same restriction on homotopies. Recall that we can think of any homotopy as a continuously

---

1One natural exception to this rule is the flat torus, which is the metric space obtained by gluing opposite sides of the unit square (or any other parallelogram) in the plane. Homotopy testing on the flat torus is nearly trivial.
deforming curve. Even though the deforming curve starts and ends on the vertices and edges of \(\Sigma\), it may pass through faces in some or all intermediate stages.

However, just as with generic curves in the plane, any homotopy between traversal curves in surface maps can be decomposed into a finite sequence of combinatorial moves, of two types:

- A spur move either inserts or deletes a subpath consisting of a dart followed by its reversal; we call such a subpath a spur (or sometimes a spike).
- A face move replaces a subpath of the boundary of some face \(f\) with the complementary subpath on the boundary of \(f\).

Thus, asking whether a closed walk \(W\) in some map \(\Sigma\) is contractible is a purely combinatorial problem: Is there a sequence of spur and face moves that transforms \(W\) into a trivial walk?

### 21.3 Reducing to a System of Loops

Let \(W\) be a given closed walk through some surface map \(\Sigma\). Like every other surface-map algorithm, we begin by computing an arbitrary tree-cotree decomposition \((T, L, C)\) of \(\Sigma\). We then reduce \(\Sigma\) to a system of \(\bar{g}\) loops \(\Lambda = \Sigma / T \setminus C\) by contracting every edge in \(T\) and then deleting every edge in \(C\). simultaneously modifying the closed walk \(W\) as follows:

- When we contract edges in \(T\), we simply remove any darts of \(T\) from \(W\). After all edges in \(T\) are contracted, all remaining edges are loops.
- Then when we delete edges in \(C\), we replace each dart in \(W\) whose edge is in \(C\) with a walk around the boundary of the unique face of \(\Lambda\), as shown in the figure below.

The reduction takes \(O(n + \ell') = O(n + \bar{g}\ell)\) time to compute, where \(n\) is the complexity of the
input map $\Sigma$, $\ell$ is the length of the input walk $W$ in $\Sigma$, and $\ell'$ is the length of the transformed walk $W'$ in $\Lambda$. (Later I'll remove the factor of $\bar{g}$ by reducing to a slightly different map.)

We have now reached the special case of the homotopy problem that Dehn actually solved in 1911: Given a closed walk in a system of loops, is there a sequence of spur and face moves that reduce it to a trivial walk?

### 21.4 Universal Cover

The universal cover $\tilde{\Lambda}$ of $\Lambda$ is an infinite planar map obtained by gluing an infinite lattice of copies of the single face of $\Lambda$ along corresponding edges. Combinatorially, $\tilde{\Lambda}$ is isomorphic to a regular tiling of the hyperbolic plane by regular $2\bar{g}$-gons meeting $2\bar{g}$ at each vertex. For example, if the input map $\Sigma$ is an orientable map with genus 2, we reduce $\Sigma$ to a system of loops $\Lambda$ contains four loops, whose universal cover $\tilde{\Lambda}$ is an 8-regular hyperbolic tiling of octagons.

![Figure 3: The universal cover of an orientable system of loops with genus 2.](image-url)

Formally, a covering map is a continuous surjection $\pi: X' \rightarrow X$ between topological spaces, such that each point $x \in X$ lies in an open neighborhood $U \subset X$ whose preimage $\pi^{-1}(U)$ is the disjoint union of open sets $\bigcup_{i \in I} U_i$, where the restriction of $\pi$ to each set $U_i$ is a homeomorphism to $U$. Space $X'$ is called a covering space of $X$ if there is a covering map from $X'$ to $X$. By convention, covering spaces are assumed to be connected.

Covering maps can also be formulated combinatorially as follows. A map-covering map is a surjective function $\pi: \Sigma' \rightarrow \Sigma$ between surface maps that sends vertices to vertices, darts to darts, and faces to faces, and that preserves degrees of vertices and faces. For example, if the maps $\Sigma'$ and $\Sigma$ are represented rotation systems $(D', \text{rev}', \text{succ}')$ and $(D, \text{rev}, \text{succ})$, a covering map is a function $\pi: D' \rightarrow D$ such that $\text{rev} \circ \pi = \pi \circ \text{rev}$ and $\text{succ} \circ \pi = \pi \circ \text{succ}'$, and $\pi$ sends every orbit of $\text{succ}'$ or $\text{rev}'(\text{succ}')$ bijectively to an orbit of $\text{succ}$ or $\text{rev}(\text{succ})$, respectively. There is a similar combinatorial formulation for reflection systems.

A lift of any vertex $v$ of $\Sigma$ to a covering space $\Sigma'$ is a vertex $v'$ of $\Sigma'$ such that $\pi(v') = v$. Lifts of darts, edges, and faces are defined similarly. Locally, it is impossible to distinguish between a feature of $\Sigma$ and any of its lifts in $\Sigma'$.

The universal cover of a space $X$ is the unique (connected) covering space $\tilde{X}$ that is simply connected, meaning all closed curves are contractible. The universal cover $\tilde{X}$ is also the “largest”
(connected) covering space of \( X \), meaning \( \tilde{X} \) is a covering space of every (connected) covering space of \( X \). For almost all surfaces, the universal cover is homeomorphic to the plane. (The only exceptions are the sphere, which is its own universal cover, and the projective plane, whose universal cover is the sphere.) Similarly, universal cover of any surface map \( \Sigma \) is a spherical map if \( \Sigma \) lies on the sphere or the projective plane, and an infinite planar map otherwise.

**Lemma:** A closed walk \( W \) in a surface map \( \Sigma \) is contractible if and only if \( W \) is the projection of a closed walk \( \tilde{W} \) in the universal cover \( \tilde{\Sigma} \).

**Proof:** Let \( \pi : \tilde{\Sigma} \to \Sigma \) denote the universal covering map.

First, let \( \tilde{W} \) be any closed walk in \( \tilde{\Sigma} \), and let \( W = \pi(\tilde{W}) \). Because \( \tilde{\Sigma} \) is simply connected, \( \tilde{W} \) must be contractible. Consider any sequence of spur moves and face moves \( \tilde{W}_0 \to \tilde{W}_1 \to \tilde{W}_2 \to \cdots \to \tilde{W}_N \) that reduces \( \tilde{W} = \tilde{W}_0 \) to a trivial closed walk \( \tilde{W}_N \). For each index \( i \), let \( W_i = \pi(\tilde{W}_i) \). Then \( W_0 \to W_1 \to W_2 \to \cdots \to W_N \) is also a sequence of spur moves and face moves that reduces \( W = W_0 \) to a trivial walk \( W_N \). Specifically, if \( \tilde{W}_{i-1} \to \tilde{W}_i \) is a spur move on some edge \( \tilde{e} \), then \( W_{i-1} \to W_i \) is a spur move on the edge \( \pi(\tilde{e}) \), and if \( \tilde{W}_{i-1} \to \tilde{W}_i \) is a face move on some edge \( \tilde{f} \), then \( W_{i-1} \to W_i \) is a face move on the edge \( \pi(\tilde{f}) \). We conclude that \( W \) is contractible.

Conversely, let \( W \) be any closed walk in \( \Sigma \). Formally, \( W \) is an alternating sequence \( v_0, d_1, v_1, d_2, \ldots, d_t, v_t \) of vertices and darts, where \( v_t = v_0 \) and for each index \( i \), we have \( v_{i-1} = \text{tail}(d_i) \) and \( v_i = \text{head}(d_i) \). We can iteratively lift \( W \) to a (not necessarily closed) walk \( \tilde{W} \) as follows. First, let \( \tilde{v}_0 \) be any lift of \( v_0 \), and then for each index \( i > 0 \), let \( \tilde{d}_i \) and \( \tilde{v}_i = \text{head}(\tilde{d}_i) \). Now suppose \( W \) is contractible. Then there is a sequence \( W_0 \to W_1 \to W_2 \to \cdots \to W_N \) of spur and face moves transforming some trivial walk \( W_0 \) into \( W_N = W \). (Yes, we are expanding here rather than contracting.) Fix an arbitrary lift \( \tilde{W}_0 \) of \( W_0 \). Then for each index \( j \), let \( \tilde{W}_j \) be the closed walk in \( \tilde{X} \) obtained from \( \tilde{W}_{j-1} \) by lifting the move \( W_{j-1} \to W_j \). Specifically, if \( W_{j-1} \to W_j \) inserts a spur \( v_i \to w \to v_i \) into \( W_{j-1} \) at its \( i \)th vertex \( v_i \), then \( \tilde{W}_j \) is obtained by inserting a spur \( \tilde{v}_i \to \tilde{w} \to \tilde{v}_i \) into \( \tilde{W}_{j-1} \) at its \( i \)th vertex \( \tilde{v}_i \), where \( \tilde{v}_i \to \tilde{w} \) is the unique lift of dart \( v_i \to w \) whose tail is \( \tilde{v}_i \). Face moves can be lifted similarly. By induction, each of the resulting walks \( \tilde{W}_j \) is closed, and therefore the final walk \( \tilde{W}_N \) is a closed walk such that \( \pi(\tilde{W}_N) = W \). \( \square \)

With this lemma in hand, we can now phrase the contractibility problem in the form that Dehn' considered it. Given a closed walk \( W \) in some system of loops \( \Lambda \), is \( W \) the projection of a closed walk in the universal cover \( \tilde{\Lambda} \)?

### 21.5 Dehn’s Lemma

We use a version of the combinatorial Gauss-Bonnet theorem for surfaces with boundary (some faces marked as missing). Here curvature is defined as

\[
\kappa(f) = 1 - \sum_{c \in f} \angle c \quad \kappa(v) = 1 - \frac{1}{2} \deg(v) + \sum_{c \in v} \angle c
\]

where \( \deg(v) \) is the number of darts incident to \( v \), not the number of corners. Then as usual we have \( \sum_v \kappa(v) + \sum_f \kappa(f) = \chi \). In our application, we will always set \( \angle c = 1/4 \), so

\[
\kappa(f) = 1 - \frac{1}{4} \deg(f) \quad \kappa(v) = 1 - \frac{1}{2} \deg(v) + \frac{1}{4} \deg_2(v)
\]

\end{rawtext}

4
where $\deg_2(v)$ is the number of corners incident to $v$.

**Lemma:** Every nontrivial closed walk $\tilde{W}$ in $\tilde{\Lambda}$ contains either a spur or at least $2\bar{g} - 2$ edges on the boundary of some face.

**Proof:** To keep things simple, consider the special case where $\tilde{W}$ is a nontrivial simple closed curve; in particular, $\tilde{W}$ has no spurs. Let $\Delta = (V, E, F)$ denote the map of the disk consisting of faces inside $\tilde{W}$. Assign every corner of $\Delta$ the angle $1/4$.

- For any face $f$, we have $\kappa(f) = 1 - \deg(f)/4 = 1 - \bar{g} < 0$.
- For any interior vertex $v$, we have $\deg_2(v) = \deg(v)$ and therefore $\kappa(v) = 1 - \deg(v)/4 = 1 - \bar{g} < 0$.
- For any boundary vertex $v$, we have $\deg_2(v) = \deg(v) - 1$ and therefore $\kappa(v) = 3/4 - \deg(v)$.

Call a boundary vertex of $\Delta$ convex if it is incident to exactly one face of $\Delta$. Every convex boundary vertex has degree 2 in $\Delta$ and therefore has curvature $1/4$; all other vertices of $\Delta$ have curvature at most 0.

The combinatorial Gauss-Bonnet theorem implies that $\sum_v \kappa(v) + \sum_f \kappa(f) = 1$, which implies that

$$|F|(1 - \bar{g}) + |V_+|/4 \geq 1 \implies |V_+| \geq (2\bar{g} - 4)F + 1$$

where $V_+$ is the set of convex vertices. It follows that some face $f$ is incident to $2\bar{g} - 3$ convex boundary vertices. These must be consecutive around the boundary of $f$. We conclude that $f$ has $2\bar{g} - 2$ consecutive edges on the boundary of $\Delta$. \hfill \square

Projection back to the system of loops immediately gives us the following corollary.

**Corollary:** Every nontrivial contractible closed walk $W$ in $\Lambda$ contains either a spur or at least $2\bar{g} - 2$ edges on the boundary of some face.

### 21.6 Dehn’s algorithm!

Finally, Dehn’s algorithm uses a simple greedy improvement strategy: Repeatedly remove spurs (using spur moves) and long boundary subpaths (using face moves) from $W'$ until no more remain, and then return true if and only if the remaining walk is trivial. Correctness follows immediately from Dehn’s lemma.

To find long boundary subpaths efficiently, we assign a unique label to each dart and represent $W$ as a (circular) string of dart labels, sorted in a circular linked list. Then we slide a window of length $2\bar{g} - 2$ over the string, checking each of the $O(\bar{g})$ possible long boundary subpaths at each position. Using brute force string comparisons, this check takes $O(\bar{g}^2)$ time per position. We can improve this to $O(1)$ time per position by building a DFA that matches all long boundary subpaths; building this DFA adds $O(\bar{g}^2)$ time to preprocessing.

Whenever we find a long boundary subpath, we replace it with its complement (of length 2) and move the window back $2\bar{g}$ steps; we charge both the deletion and the time to find the long boundary subpath to the decrease in string length. Similarly, whenever we notice a spur, we can remove it in $O(1)$ time. The algorithm ends after $O(\ell')$ iterations.

Thus, the overall running time of Dehn’s algorithm, starting with an arbitrary surface map $\Sigma$ with complexity $n$ and Euler genus $\bar{g}$, is $O(n + \bar{g}^2 + \bar{g}\ell')$. 
21.7 System of quads

To remove the dependence on $\bar{g}$ in the running time, we reduce to a different elementary map called a system of quads.

For any surface map $\Sigma = (V, E, F)$, the radial map $\Sigma^\circ = (V^\circ, E^\circ, F^\circ)$ is defined as follows:

- $V^\circ = V \cup F^*$, where $f^*$ is the set of vertices of the dual map $\Sigma^*$.
- Edges in $E^\circ$ correspond to corners in $\Sigma$. Any primal vertex $v$ and dual vertex $f^*$ are connected by one radial edge for each incidence between $v$ and $f$.
- Faces in $F^\circ$ correspond to edges in $\Sigma$. Every face $e^\circ \in F^\circ$ has degree 4; its vertices are the endpoints of $e$ and the endpoints of the dual edge $e^*$.

The radial map (black) of a surface map (white).

Figure 4: The radial map (black) of a surface map (white).

A system of quads is the radial map of a system of loops: $Q = \Lambda^\circ$. This map has exactly 2 vertices, $2\bar{g}$ edges, and $\bar{g}$ quadrilateral faces.

Figure 5: A system of quads on an orientable surface of genus 2 (and thus Euler genus 4). Each face has a unique color.

We can reduce an arbitrary closed walk $W$ in an arbitrary map $\Sigma$ to a homotopic closed walk $W'$ in a system of quads $Q$ by modifying our earlier reduction to a system of loops. Fix an arbitrary tree-cotree decomposition $(T, L, C)$, and contract the edges in the spanning tree $T$. Let $v$ and $f$ respectively denote the only vertex and the only face of the system of loops $\Lambda = \Sigma \setminus T / C$. Each edge $e$ of $\Sigma \setminus T$ can be considered (or perturbed into) a path through $f$ from one corner to another. We replace each such edge with the corresponding path of length 2 in $Q = \Lambda^\circ$, from $v$ to $f^*$ to $v$. The resulting walk $W'$ in $Q$ has length at most $2\ell$, and the reduction requires $O(n + \ell)$ time.

The universal cover of $Q$ is a hyperbolic tiling by squares meeting $2\bar{g}$ at each vertex. Our earlier arguments imply that $W'$ (and therefore $W$) is contractible if and only if $W'$ is the projection of a closed walk in the universal cover $\tilde{Q}$.

21.8 Brackets

Dehn’s lemma still applies to the infinite hyperbolic tiling $\tilde{Q}$—Every closed walk in $\tilde{Q}$ contains either a spur of a subpath that covers all but two edges of some face. But now the complement of a “long” boundary subpath also has length 2; a single face move does not necessarily decrease the length of the walk. We need to find larger moves that are still simple enough to find and execute quickly, but that are guaranteed to shrink any closed walk.
Figure 6: Reducing to a system of quads. (Pairs of triangles with the same color comprise faces.)

Figure 7: The universal cover of an orientable system of quads with genus 2.
To make this easier, we encode the walk $W'$ as a turn sequence. The turn of any subwalk $u \to v \to w$ of $W$ is the number of corners at the middle vertex $v$ to the left of that subpath, modulo $\bar{g}$. Thus, for example, a spur is any subpath of $W$ with turn 0. The regularity of the tiling $\bar{Q}$ implies that the contractibility of any closed walk depends only on its (cyclic) turn sequence. Moreover, we can easily compute the turn sequence of any walk in time proportional to its length.

A right bracket is any subpath whose turn sequence consists of 1, followed by zero or more 2s, followed by 1. A left bracket is any subpath whose turn sequence has the form $-1,-2,\ldots,-2,-1$, for any number of $-2$s. (In the interest of readability, from now on I will indicate negation with a bar instead of a minus sign; for example, $\bar{2} = -2$.)

**Bracket figure**

**Lemma:** Every nontrivial contractible closed walk in $Q$ contains a spur or a bracket.

**Proof:** [Combinatorial Gauss-Bonnet again.]

The previous lemma implies that we can reduce any nontrivial contractible closed walk in $Q$ either using a spur move or by “sliding” a bracket. Both of these moves can be performed entirely by modifying the turn sequence. For example, removing a spur preceded by turn $x$ and followed by turn $y$ leaves a single turn with value $x + y$. (All turn arithmetic is implicitly performed modulo $\bar{g}$.)

$$x, 0, y \rightarrow x + y$$

$$x, 1, 2^k, 1, y \rightarrow x - 1, \bar{2}^k, y - 1$$

$$x, \bar{1}, \bar{2}^k, \bar{1}, y \rightarrow x + 1, 2^k, y + 1$$

Here superscripts represent repetition, not exponentiation.

[[bracket slide figure]]

### 21.9 Reduction algorithm

To make detecting and sliding brackets easier, we actually store and manipulate a run-length encoding of the turn sequence. Instead of recording repeated turns explicitly, we store a sequence of pairs $((\tau_0, r_0), (\tau_1, r_1), \ldots)$ to represent $r_0$ repetitions of turn $\tau_0$, followed by $r_1$ repetitions of $\tau_1$, and so on. Thus, any bracket turn sequence overlaps at most five runs. (In fact, it suffices to encode only runs of 2s and $\bar{2}$s.)

We now proceed as in Dehn’s classical algorithm. We slide a window of width 5 over the run-length-encoded turn sequence; whenever the window contains a spur or a bracket, we modify the runs within the window to perform a spur move or bracket move, and then move the window back five steps. At each window position, we need $O(1)$ time to detect spurs and brackets, and $O(1)$ time to perform each spur of bracket move. The algorithm iterates until we have made a complete scan with no reductions, in which case we are done, or at most five runs remain in the run sequence, in which case we can complete the algorithm in $O(1)$ additional time. Standard amortisation arguments imply that the reduction algorithm runs in $O(\ell)$ time.

Thus, the overall running time of Dehn’s algorithm, starting with an arbitrary surface map $\Sigma$ with complexity $n$, is $O(n + \ell)$, with no hidden dependence on the surface genus.

### 21.10 Sir Not

- surfaces with boundary
• projective plane, torus, and Klein bottle
• testing free homotopy between cycles
• isotopy testing for graph embeddings [Ladegaillerie 74] [Colin de Verdière and De Mesmay 14]