Finding one tight cycle

SERGIO CABELLO
University of Ljubljana and IMFM
MATT DEVOS
Simon Fraser University
JEFF ERICKSON
University of Illinois at Urbana-Champaign
and
BOJAN MOHAR
Simon Fraser University

A cycle on a combinatorial surface is tight if it as short as possible in its (free) homotopy class. We describe an algorithm to compute a single tight, non-contractible, essentially simple cycle on a given orientable combinatorial surface in $O(n \log n)$ time. The only method previously known for this problem was to compute the globally shortest non-contractible or non-separating cycle in $O(\min\{g^3, n\}\, n \log n)$ time, where g is the genus of the surface. As a consequence, we can compute the shortest cycle freely homotopic to a chosen boundary cycle in $O(n \log n)$ time, a tight octagonal decomposition in $O(gn \log n)$ time, and a shortest contractible cycle enclosing a non-empty set of faces in $O(n \log^2 n)$ time.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems; G.2.2 [**Discrete Mathematics**]: Graph Theory—*Graph Algorithms*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: combinatorial surface, non-contractible cycle, non-separating cycle, topological graph theory

---

Research partially supported by the European Community Sixth Framework Programme under a Marie Curie Intra-European Fellowship, by an NSERC Discovery Grant (Canada), by the Canada Research Chair program, and by the Slovenian Research Agency, project J1–7218 and program P1–0297.
A preliminary version of this article appeared in Proceedings of the 19th ACM-SIAM Symposium on Discrete Algorithms, ACM, 2008, pp. 527-531.
Authors' address: Sergio Cabello, IMFM & FMF, Department of Mathematics, University of Ljubljana, Slovenia, sergio.cabello@fmf.uni-lj.si. Matt DeVos, Department of Mathematics, Simon Fraser University, mdevos@sfu.ca. Jeff Erickson, Department of Computer Science, University of Illinois at Urbana-Champaign, jeffe@cs.uiuc.edu. Bojan Mohar, Department of Mathematics, Simon Fraser University, mohar@sfu.ca; on leave from IMFM & FMF, Department of Mathematics, University of Ljubljana, Slovenia.

1. INTRODUCTION

Cutting along curves is the basic tool for working with topological surfaces. When the surface is equipped with a metric, the surgery is typically made along shortest non-trivial cycles, where non-trivial may mean non-contractible or (surface) non-separating, depending on the application. Here, we are interested in cycles with a different property: a cycle is *tight* if it is shortest in its free homotopy type. Note that a shortest non-trivial cycle is going to be tight, but the converse does not hold.

We are interested in the algorithmic aspects of finding a tight, non-trivial cycle. Like most previous algorithmical works concerning curves on surfaces [Cabello and Chambers 2007; Cabello and Mohar 2007; Chambers et al. 2006; Colin de Verdière and Erickson 2006; É. Colin de Verdière and F. Lazarus 2004; 2005; Eppstein 2003; Erickson and Har-Peled 2004; Erickson and Whittlesey 2005; Kutz 2006], we consider the *combinatorial surface* model. A combinatorial surface $\mathcal{M}$ is an edge-weighted multigraph G embedded on a surface, and only curves arising from walks in G are considered. The length of a path is the sum of the weights of its edges, counted with multiplicity. The complexity of a combinatorial surface, denoted by n, is the sum of the number of its vertices, edges, and faces.

The theory of graphs embedded on surfaces, a natural generalization of the theory of planar graphs, is a very active research area. See the monograph [Mohar and Thomassen 2001] for an introduction. Algorithmical aspects of topological graph theory are also playing an important role in several graph problems. See for example the recent linear-time algorithm of Kawarabayashi and Reed [2007] for testing if a given graph has bounded crossing number.

The main result of this paper is an algorithm to compute a tight, surface non-separating cycle on an orientable combinatorial surface in $O(n \log n)$ time. The best previous solution to the problem of finding a tight, non-trivial cycle was to compute the globally shortest non-trivial cycle, which can be done in $O(n^2 \log n)$ time with an algorithm by Erickson and Har-Peled [2004] or in $O(g^3 n \log n)$ time with an algorithm by Cabello and Chambers [2007]. (See [Cabello and Mohar 2007; Kutz 2006] for other relevant results.)

This new algorithm has the following implications:

— In the approach of [Colin de Verdière and Erickson 2006] for finding shortest curves homotopic to a given one, the bottleneck of the preprocessing part was to find a tight, non-trivial cycle. With our result, we can speed up their preprocessing from $O(\min\{g^3, n\} n \log n)$ to $O(gn \log n)$.

— We can compute the shortest cycle (freely) homotopic to a given boundary component in $O(n \log n)$ time. The previous best algorithm [Colin de Verdière and Erickson 2006] used $O(gn \log n)$ time.

— We can compute a shortest contractible cycle that encloses a non-empty set of faces in $O(n \log^2 n)$ time.

— We show that a subquadratic algorithm to find a shortest non-contractible cycle would imply a subquadratic algorithm to compute the girth of any sparse graph $G(V, E)$ with $|E| = O(|V|)$.

— In topological graph theory, several of the proofs based on cutting along shortest non-trivial cycles carry out if instead we cut along a tight, non-trivial cycle.

Thus, algorithmic counterparts of several basic theorems can be improved with our new result.

2. BACKGROUND

Surfaces. We summarize some basic concepts of topology. See [Hatcher 2001; Massey 1967; Stillwell 1993] for a comprehensive treatment.

A (topological) *surface* (or 2-manifold) Σ is a compact topological space where each point has a neighbourhood homeomorphic to the plane or to a closed halfplane. A boundary point in Σ is a point with the property that no neighbourhood is homeomorphic to the plane. The *boundary* of Σ is the union of all boundary points, and it is known to consist of a finite number (possibly 0) of connected components, each component homeomorphic to a circle. The surface is *non-orientable* if it contains a subset homeomorphic to the Möbius band, and *orientable* otherwise. An orientable surface is homeomorphic to a sphere with a number $g \geq 0$ of handles attached to it and a number $b \geq 0$ of disjoint open disks removed, for a unique pair $g, b \geq 0$. A non-orientable surface is homeomorphic to the connected sum of g projective planes and a number $b \geq 0$ of disjoint open disks removed, for a unique pair $g, b \geq 0$. In both cases, g is the *genus* of the surface and b is the number of boundary components.

A *path* in Σ is a continuous mapping $p : [0,1] \to \Sigma$, a *cycle* is a continuous mapping $\gamma : \mathbb{S}^1 \to \Sigma$, a *loop* with basepoint x is a path such that $x = p(0) = p(1)$, and an *arc* is a path whose endpoints are on the boundary. *Curve* is a generic term used for paths, cycles, arcs, and loops. A curve is *simple* when the mapping is injective, except for the common endpoint in the case of loops.

Two paths or arcs p, q with $p(0) = q(0)$ and $p(1) = q(1)$ are *homotopic* if there is a continuous function $h : [0,1]^2 \to \Sigma$ such that $p(\cdot) = h(0, \cdot)$, $q(\cdot) = h(1, \cdot)$, $h(\cdot, 0) = p(0)$, and $h(\cdot, 1) = p(1)$. Two cycles α, β are *(freely) homotopic* if there is a continuous function $g : [0,1] \times \mathbb{S}^1 \to \Sigma$ such that $\alpha(\cdot) = g(0, \cdot)$ and $\beta(\cdot) = g(1, \cdot)$. Simple curves are typically identified with their image because, up to reversal of the parameterization, any two parameterizations with the same image correspond to homotopic curves.

A cycle is *contractible* if it is homotopic to the constant loop. Cutting along a simple contractible cycle gives two connected components, and one of them is a topological disk. A simple cycle α is *non-separating* if cutting the surface along (the image of) α gives rise to a unique connected component. Non-separating cycles are non-contractible, while contractible cycles are separating. Being contractible or separating is a property invariant under homotopy of cycles.

We use the notation $\Sigma ✄ \alpha$ to denote the surface obtained after cutting Σ along a simple curve α. Points along the curve α become boundary points in $\Sigma ✄ \alpha$. Contrary to what is common in topological graph theory, we do not paste disks to the new boundary components, unless explicitly mentioned. We denote by $\Sigma ✄ (\alpha_1, \dots, \alpha_k)$ the surface obtained inductively as $(\Sigma ✄ (\alpha_1, \dots, \alpha_{k-1})) ✄ \alpha_k$.

Combinatorial surface. Most of our results will be phrased in the combinatorial surface model. This model is dual to the cross-metric surface model; see [Colin de Verdière and Erickson 2006] for a discussion. A *combinatorial surface* $\mathcal{M}$ is a surface $\Sigma(\mathcal{M})$ together with a multigraph $G(\mathcal{M})$ embedded on $\Sigma(\mathcal{M})$ so that each

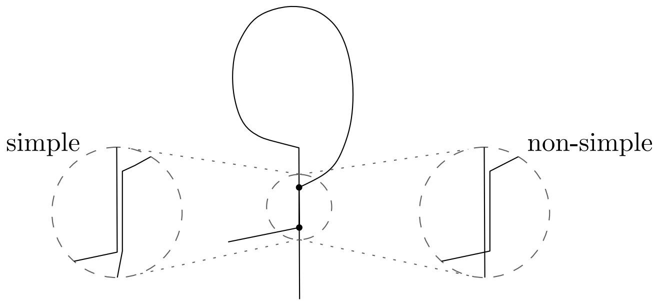

Fig. 1. Example to clarify the concept of essentially simple curve. Depending on how the walk behaves inside the disk the curve is essentially simple or not.

face of G is a topological disk. The complexity of a combinatorial surface $\mathcal{M}$ is defined as the sum of the number of vertices, edges, and faces of $G(\mathcal{M})$. The genus and the number of boundary components of $\mathcal{M}$ are those of $\Sigma(\mathcal{M})$.

In the combinatorial surface model, we only consider curves that arise as walks in $G(\mathcal{M})$. Note that a cycle in a combinatorial surface corresponds to a closed walk in $G(\mathcal{M})$, possibly with repeated edges or vertices. The *multiplicity* of a curve α is the maximum number of times that an edge appears in the graph-walk that defines α. The *complexity* of a curve α is the number of edges, counted with multiplicity, in the graph-walk that defines α.

A curve in a combinatorial surface is *essentially simple* when there is an infinitesimal continuous perturbation in Σ that makes it simple. Note that an essentially simple curve in a combinatorial surface may use the same edge multiple times. See Figure 1. Figure 6(left) also contains an example of an essentially simple curve that uses the same vertex twice. Two curves α and β in a combinatorial surface cross c times if: (i) there exist infinitesimal continuous perturbations of α and β that cross transversally c times; and (ii) any infinitesimal continuous perturbations of α and β have at least c points in common.

We assume that the graph $G(\mathcal{M})$ has positive edge-weights, which gives a "metric" to the model. The length $|\alpha|$ of a curve α is defined as the sum of the weights of the edges in the graph-walk that defines α, counted with multiplicity. A cycle or an arc is *tight* if it is shortest in its homotopy class.

Families of curves. We say that two curves α, β *include a bigon* if there are essentially simple subpaths $p_\alpha \subseteq \alpha$ and $p_\beta \subseteq \beta$ with common endpoints such that p_α and p_β bound a topological disk.

A *tight system of disjoint arcs* in a combinatorial surface with boundary is a family of essentially simple arcs $\alpha_1, \alpha_2, \ldots, \alpha_k$ such that

—no two distinct arcs α_i, α_j share an edge or cross;

—the arc α_i is a tight arc in $\mathcal{M}$✄$(\alpha_1, \ldots, \alpha_{i-1})$.

A *tight octagonal decomposition* of a surface is the decomposition induced by a family of essentially simple tight cycles $\alpha_1, \ldots, \alpha_k$, where each vertex is adjacent to at most two cycles and the boundary of each component in $\mathcal{M}$✄$(\alpha_1, \ldots, \alpha_k)$ has precisely 8 segments from the cycles $\alpha_1, \ldots, \alpha_k$. Tight octagonal decompositions

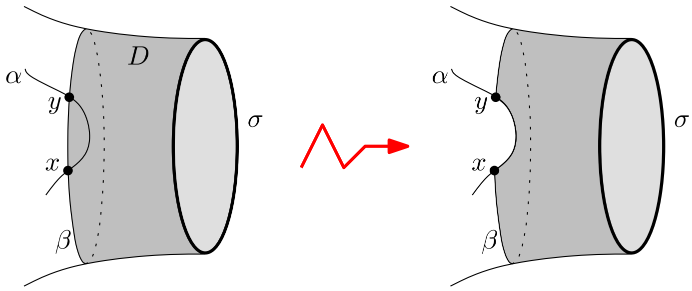

Fig. 2. Figure for Lemma 3.3. If α enters D, then β does not define a smallest cylinder D.

play a fundamental role in [Colin de Verdière and Erickson 2006] for computing shortest curves homotopic to a given curve; we refer the reader to [Colin de Verdière and Erickson 2006] for details.

3. TOOLBOX

We next list results that will be used in our proofs and algorithms.

LEMMA 3.1 [HASS AND SCOTT 1985]. *Given two homotopic cycles α, β in an orientable surface, if they have some common point, then they include a bigon.*

LEMMA 3.2 [ERICKSON AND HAR-PELED 2004]. *For any given basepoint x in a combinatorial surface, orientable or not, we can find in $O(n \log n)$ time a shortest non-separating loop with basepoint x.*

LEMMA 3.3. *Let $\mathcal{M}$ be a surface, orientable or not, with at least one boundary component, let σ be one of its boundary components, and let α be a tight essentially simple cycle or a tight essentially simple arc whose endpoints are not in σ. Every tight cycle homotopic to σ in $\mathcal{M} \setminus\!\!\setminus \alpha$ is also a tight cycle homotopic to σ in $\mathcal{M}$.*

PROOF. Let β be a tight cycle in $\mathcal{M}$ homotopic to σ. Then σ and β bound a cylinder D in $\mathcal{M}$. We choose β such that D is smallest possible, i.e., no other tight cycle homotopic to σ bounds a cylinder which is contained in D. We will show that α is disjoint from the interior of D, which will then imply that β is also a tight cycle in $\mathcal{M}' = \mathcal{M} \setminus\!\!\setminus \alpha$ homotopic to σ. To see this, suppose that α enters D; see Figure 2. Then $\alpha \cap D$ contains an essentially simple path α' whose endpoints x, y are on β, and thus α and β include a bigon. Because of tightness of β and α, both segments of this bigon have the same length, and we can replace the segment of β with α'. The new curve is homotopic to σ and contradicts the minimality of D. This completes the proof. □

LEMMA 3.4. *Let $\mathcal{M}$ be a combinatorial surface, orientable or not, with complexity n and exactly one boundary component σ. We can find in $O(n \log n)$ time a tight system of disjoint arcs $\alpha_1, \ldots, \alpha_k$ such that $\mathcal{M} \setminus\!\!\setminus (\alpha_1, \ldots, \alpha_k)$ is a topological disk of complexity $O(n)$.*

PROOF. Contract σ to a point p_σ to obtain a combinatorial surface $\mathcal{M}'$. Let $k = 2g$ if $\mathcal{M}$ is orientable and $k = g$ if $\mathcal{M}$ is non-orientable, where g is the genus of $\mathcal{M}$. Consider in $\mathcal{M}'$ a greedy system of loops $\ell_1, \ldots, \ell_k$ with basepoint p_σ, defined

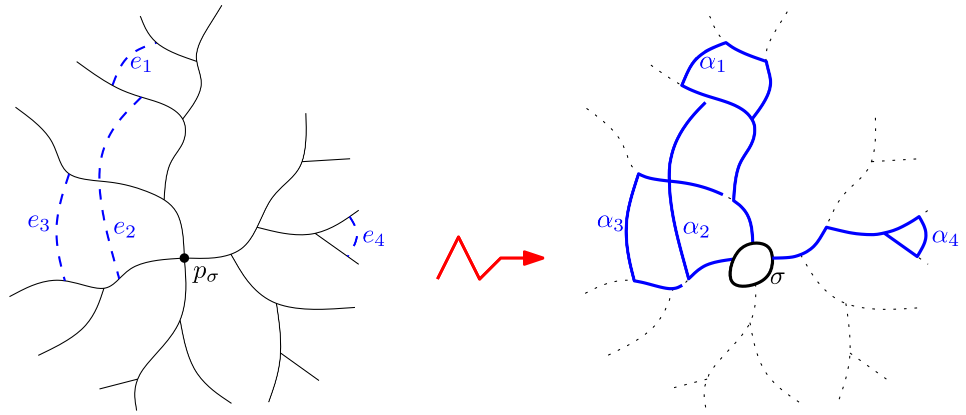

Fig. 3. Figure for Lemma 3.4. Left: Example for $k = 4$ showing T and $e_1, \dots, e_4$ as a compact representation of the loops $\ell_1, \dots, \ell_4$. Right: The final tight system of disjoint arcs $\alpha_1, \dots, \alpha_4$ obtained in the example; the crossing between α_2, α_3 does not occur in the surface.

iteratively as follows: for each i, ℓ_i is a shortest loop ℓ with basepoint p_σ such that $\mathcal{M}' \setminus\!\!\setminus (\ell_1, \dots, \ell_{i-1}, \ell)$ is connected. Erickson and Whittlesey [2005] describe an algorithm to compute in $O(n \log n)$ time a compact representation of this greedy system of loops $\ell_1, \dots, \ell_k$ in $O(n)$ space. The compact representation is given by a shortest path tree T rooted at p_σ and a collection of k edges $e_1, \dots, e_k$ not contained in T. In this representation, ℓ_i is the loop obtained by following the path in T from p_σ to an endpoint of e_i, the edge e_i, and the path in T from the other endpoint of e_i to p_σ. See Figure 3 left. Note that each loop ℓ_i has multiplicity at most two.

Unmaking the contraction back to $\mathcal{M}$, each loop ℓ_i becomes an arc β_i in $\mathcal{M}$ with endpoints at σ, and moreover $\mathcal{M} \setminus\!\!\setminus (\beta_1, \dots, \beta_k)$ is a topological disk. It follows from the greediness of the construction that each β_i is tight in $\mathcal{M} \setminus\!\!\setminus (\beta_1, \dots, \beta_{i-1})$. Note that an edge could appear in several curves β_i. However, an arc β_i intersects the union $\beta_1 \cup \dots \cup \beta_{i-1}$ in a connected subpath, namely in a subpath of T. Therefore, assigning each edge to the curve β_i with smallest index i where it appears, and removing it from the rest of curves, we obtain a set of curves $\alpha_1, \dots, \alpha_k$ that do not share any edge. See Figure 3 right. Note that this operation can be done in $O(n)$ from the implicit representation of the greedy system of loops. After this operation, each curve α_i is an essentially simple tight arc in $\mathcal{M} \setminus\!\!\setminus (\alpha_1, \dots, \alpha_{i-1})$. Since each curve α_i has multiplicity at most two and no two curves share an edge, the surface $\mathcal{M} \setminus\!\!\setminus (\alpha_1, \dots, \alpha_k)$ is a topological disk of complexity $O(n)$, as required. □

Lemma 3.5. *Let $\mathcal{M}$ be a combinatorial surface, orientable or not, with complexity n and $b \geq 2$ boundary components. We can find in $O(n \log n)$ time a tight system of disjoint arcs $\beta_1, \dots, \beta_{b-1}$ such that $\mathcal{M} \setminus\!\!\setminus (\beta_1, \dots, \beta_{b-1})$ has one boundary and complexity $O(n)$.*

Proof. Let $\sigma_1, \dots, \sigma_b$ be the boundary cycles of $\mathcal{M}$. We contract each σ_i to a point p_i, and find a shortest path tree T from p_1. This can be done in $O(n \log n)$ time. Let us re-index the points $p_2, \dots, p_b$ so that no point p_i appears in the subtree of T rooted at p_j for $j < i$; see Figure 4. Let π_i denote the shortest path from p_1

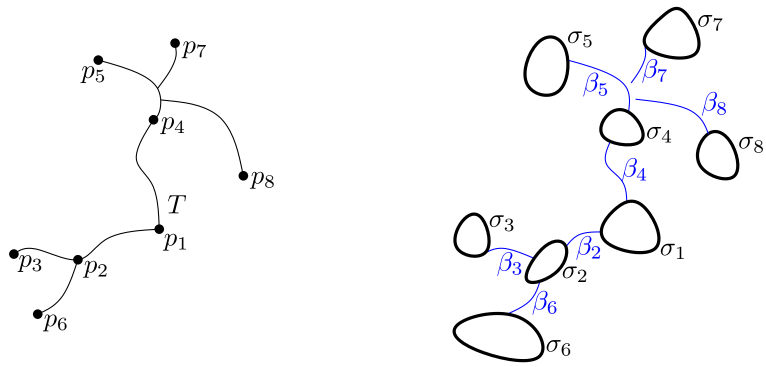

Fig. 4. Figure for Lemma 3.5. Example showing the shortest path tree T from p_1 to $p_2, \ldots, p_b$ (left) and the paths $\beta_2, \ldots, \beta_b$ (right). The paths β_7, β_8 start at β_5; we show the separation to remark that $\beta_5, \beta_7, \beta_8$ are pairwise edge-disjoint.

to p_i contained in T. Each edge can appear in several paths π_i, but we proceed like in the proof of Lemma 3.4: we assign each edge to the path π_i with smallest index that contains it, and delete it from the rest. Let $\beta_2, \ldots, \beta_b$ be the paths that are obtained. The curves $\beta_2, \ldots, \beta_b$ in the original surface $\mathcal{M}$ form a tight system of disjoint arcs with the property that $\mathcal{M}\text{✂}(\beta_2, \ldots, \beta_b)$ has one boundary. Moreover, the multiplicity of each curve β_i is one, and therefore $\mathcal{M}\text{✂}(\beta_2, \ldots, \beta_b)$ has complexity $O(n)$. □

4. FINDING ONE TIGHT CYCLE

LEMMA 4.1. *Let $\mathcal{M}$ be a combinatorial surface, orientable or not, with complexity n, $b \geq 2$ boundary components, and let σ be one of its boundary cycles. We can find in $O(n \log n)$ time a tight cycle homotopic to σ that has complexity $O(n)$.*

PROOF. Assume first that $b = 2$, and let σ' be the boundary component distinct from σ. Glue a disk over σ, and construct a tight system of disjoint arcs $\alpha_1, \ldots, \alpha_k$ as described in Lemma 3.4. Cutting the surface $\mathcal{M}$ along $\alpha_1, \ldots, \alpha_k$ leaves an annulus $\mathcal{A}$ of complexity $O(n)$ whose boundary components are σ and σ'. Furthermore, it follows from Lemma 3.3 that a tight cycle homotopic to σ in $\mathcal{A}$ is a tight cycle homotopic to σ in $\mathcal{M}$. Finally, the shortest generating cycle in $\mathcal{A}$ has complexity $O(n)$ and can be computed in $O(n \log n)$ time using the algorithm by Frederickson [1987] because $\mathcal{A}$ has linear complexity. This concludes the case when $b = 2$.

The case when $b > 2$ can be reduced to $b = 2$ as follows. We glue a disk over σ and construct a tight system of disjoint arcs $\beta_1, \ldots, \beta_{b-2}$ as described in Lemma 3.5. Note that the surface $\mathcal{M}' = \mathcal{M}\text{✂}(\beta_1, \ldots, \beta_{b-2})$ has two boundaries, one of them arising from σ, and has complexity $O(n)$. A tight cycle homotopic to the boundary σ in $\mathcal{M}'$ is a tight cycle homotopic to σ in $\mathcal{M}$ because of Lemma 3.3. Finally, note that a tight cycle homotopic to the boundary σ in $\mathcal{M}'$ can be found in $O(n \log n)$ time because $\mathcal{M}'$ has two boundaries, which was the previous case. □

Note that in the following two results we only consider *orientable surfaces.*

LEMMA 4.2. *Let $\mathcal{M}$ be an* orientable *combinatorial surface. Let ℓ_x be a shortest*

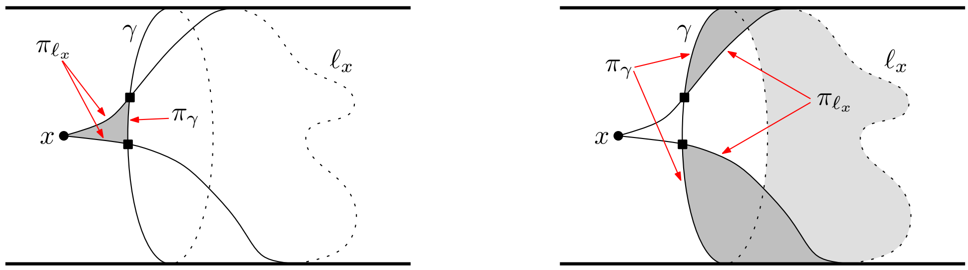

Fig. 5. Figure for Lemma 4.2. The gray region represents a bigon between ℓ_x and γ. Left: the case $x \in \pi_{\ell_x}$. Right: the case $x \in \pi_\gamma$.

non-separating loop with basepoint x, and let ℓ'_x, ℓ''_x be the two copies of ℓ_x in $\mathcal{M} \setminus\!\!\setminus \ell_x$. The tight cycle homotopic to ℓ'_x in $\mathcal{M} \setminus\!\!\setminus \ell_x$ or the tight cycle homotopic to ℓ''_x in $\mathcal{M} \setminus\!\!\setminus \ell_x$ is a tight cycle homotopic to ℓ_x in $\mathcal{M}$.

Proof. We will show that in $\mathcal{M}$ there is a cycle γ that is homotopic to ℓ_x, it is tight, and does not cross ℓ_x. Since γ does not cross ℓ_x, then γ is also homotopic to ℓ'_x or ℓ''_x in $\mathcal{M} \setminus\!\!\setminus \ell_x$, and it is tight, which implies the result.

Let γ be a tight cycle that is homotopic to ℓ_x (in $\mathcal{M}$) and crosses ℓ_x as few times as possible. We want to show that γ and ℓ_x do not cross. Assume for contradiction that γ and ℓ_x cross. Then, by Lemma 3.1, they include a bigon. Let $\pi_\gamma \subset \gamma$ and $\pi_{\ell_x} \subset \ell_x$ be the two subpaths that enclose the bigon; π_γ and π_{ℓ_x} are homotopic paths; see Figure 5. Let q_γ be the subpath $\gamma \setminus \pi_\gamma$ and let q_{ℓ_x} be the subpath $\ell_x \setminus \pi_{\ell_x}$. We distinguish two cases:

π_{ℓ_x} contains x. Let δ be the cycle π_γ concatenated with q_{ℓ_x}. Note that δ crosses ℓ_x twice less than γ does. Since π_γ and π_{ℓ_x} are homotopic, δ is homotopic to ℓ_x and γ. Since π_{ℓ_x} concatenated with q_γ is a non-separating cycle through x, it holds that

$$|\ell_x| = |\pi_{\ell_x}| + |q_{\ell_x}| \le |\pi_{\ell_x}| + |q_\gamma|,$$

which implies $|q_{\ell_x}| \le |q_\gamma|$. We conclude that

$$|\delta| = |\pi_\gamma| + |q_{\ell_x}| \le |\pi_\gamma| + |q_\gamma| = |\gamma|,$$

and since δ crosses ℓ_x twice less than γ, we get a contradiction.

π_{ℓ_x} does not contain x. Let δ be the cycle π_{ℓ_x} concatenated with q_γ. Note that δ crosses ℓ_x twice less than γ does. Since π_γ and π_{ℓ_x} are homotopic, δ is homotopic to γ and ℓ_x. Since q_{ℓ_x} concatenated with π_γ is a non-separating cycle through x, it holds that

$$|\ell_x| = |q_{\ell_x}| + |\pi_{\ell_x}| \le |q_{\ell_x}| + |\pi_\gamma|,$$

which implies $|\pi_{\ell_x}| \le |\pi_\gamma|$. We conclude that

$$|\delta| = |\pi_{\ell_x}| + |q_\gamma| \le |\pi_\gamma| + |q_\gamma| \le |\gamma|,$$

and since δ crosses ℓ_x twice less than γ, we get a contradiction.

□

Note that Lemma 4.2 does not hold for non-orientable surfaces: any two non-contractible cycles in the projective plane cross an odd number of times.

Theorem 4.3. *Let $\mathcal{M}$ be an orientable combinatorial surface with complexity n. We can find in $O(n \log n)$ time a cycle that is tight, essentially simple, surface non-separating, and has complexity $O(n)$.*

Proof. Choose a point $x \in \mathcal{M}$, and construct a shortest non-separating loop ℓ_x with basepoint x. Since $\mathcal{M}$ is an orientable surface, $\mathcal{M}' = \mathcal{M} \mathbin{\text{✂}} \ell_x$ has two new boundary components ℓ'_x and ℓ''_x arising from ℓ_x. We find γ', a tight cycle homotopic to ℓ'_x in $\mathcal{M} \mathbin{\text{✂}} \ell_x$, and γ'', a tight cycle homotopic to ℓ''_x in $\mathcal{M} \mathbin{\text{✂}} \ell_x$, and return the shorter cycle among γ', γ''. This finishes the description of the algorithm.

The cycle γ_{min} returned by the algorithm is tight because of Lemma 4.2. Since the cycle γ_{min} is homotopic to the essentially simple, non-separating loop ℓ_x in $\mathcal{M}$, it follows that γ_{min} is also non-separating and essentially simple. As for the running time, note that ℓ_x can be found in $O(n \log n)$ time because of Lemma 3.2, and the cycles γ', γ'' can also be obtained in $O(n \log n)$ time using Lemma 4.1 because $\mathcal{M} \mathbin{\text{✂}} \ell_x$ has at least two boundary components. □

It is unclear if our approach can be extended to non-orientable surfaces because Lemma 4.2 does not hold for non-orientable surfaces.

5. CONSEQUENCES AND CONCLUSIONS

Using Theorem 4.3 we can find a tight octagonal decomposition of an orientable surface $\mathcal{M}$ without boundary in $O(gn \log n)$ time, improving the previous $O(n^2 \log n)$ time bound of [Colin de Verdière and Erickson 2006]. This improves the preprocessing time in their results.

Theorem 5.1. *Let $\mathcal{M}$ be an orientable cross-metric surface with complexity n, genus $g \geq 2$, and no boundary. We can construct a tight octagonal decomposition of $\mathcal{M}$ in $O(gn \log n)$ time.*

Proof. Consider the construction described in Theorem 4.1 of [Colin de Verdière and Erickson 2006]. Their first step is to find a tight cycle in $\mathcal{M}$, which they implement finding a globally shortest non-separating cycle in $O(n^2 \log n)$ time. (Finding this cycle can be done in $O(g^3 n \log n)$ time using the more recent result of Cabello and Chambers [2007].) Using Theorem 4.3, we can now perform this first step in $O(n \log n)$ time. After this, the rest of their construction takes $O(gn \log n)$ time, and the result follows. □

Theorem 5.2. *Let $\mathcal{M}$ be an orientable combinatorial surface with complexity n, genus $g \geq 2$, and no boundary. Let p be a path on $\mathcal{M}$, represented as a walk in $G(\mathcal{M})$ with complexity k. We can compute a shortest path p' homotopic to p with complexity k' in $O(gn \log n + gk + gn\bar{k})$ time, where $\bar{k} = \min\{k, k'\}$.*
For a cycle γ, we can do the same in $O(gn \log n + gk + gn\bar{k} \log(n\bar{k}))$ time.

Proof. The preprocessing time for constructing a tight octagonal decomposition has gone down from $O(n^2 \log n)$ to $O(gn \log n)$ because of the previous result. The result follows then from the algorithms in [Colin de Verdière and Erickson 2006]. □

Another consequence of our results is a faster algorithm for computing a shortest cycle homotopic to a given boundary of a surface. For the following results, we can return to arbitrary surfaces, orientable or not.

THEOREM 5.3. *Let $\mathcal{M}$ be a combinatorial surface with complexity n, orientable or not, and let σ be a given boundary cycle in $\mathcal{M}$. We can find in $O(n \log n)$ time a tight cycle homotopic to σ that has complexity $O(n)$.*

PROOF. If $\mathcal{M}$ has more than two boundary components, the result follows from Lemma 4.1.

If $\mathcal{M}$ is orientable and has only one boundary component σ, we compute a tight non-separating, essentially simple cycle γ using Theorem 4.3, and then find a tight cycle $\tilde{\sigma}$ homotopic to σ in $\mathcal{M} ✂ \gamma$. Finally, we return the cycle $\tilde{\sigma}$. This algorithm is correct because the returned cycle $\tilde{\sigma}$ is homotopic to σ and is tight because of Lemma 3.3. As for the running time and the complexity of $\tilde{\sigma}$, note that γ is obtained in $O(n \log n)$ time and has complexity $O(n)$ because of Theorem 4.3. Therefore $\mathcal{M} ✂ \gamma$ has precisely three boundary components and complexity $O(n)$. Hence $\tilde{\sigma}$ can be obtained in $O(n \log n)$ time and has complexity $O(n)$ because of Lemma 4.1.

It remains the case when $\mathcal{M}$ is non-orientable and has only one boundary component σ. We use the *orientable double cover* $\mathcal{M}_o$ of $\mathcal{M}$, which is a particular covering space of $\mathcal{M}$. We next describe an algorithmic construction of $\mathcal{M}_o$; see [Hatcher 2001, Section 1.3] or [Massey 1967, Chapter 5] for a general treatment of covering spaces. Since each face of f is a topological disk, it has two distinct sides. Make two copies f', f'' of each face f of $\mathcal{M}$, color blue one side of f' and red the other side, and color also the sides of f'' exchanging the colors red and blue with respect to f'. Finally, for any edge e of $\mathcal{M}$ between faces f_1, f_2, glue along the copies of e either the two pairs f'_1, f'_2, and f''_1, f''_2 or the two pairs f'_1, f''_2, and f''_1, f'_2, so that there are consistent colors on either side after gluing. One then obtains the surface $\mathcal{M}_o$, which turns out to be connected and orientable. Note that $\mathcal{M}_o$ has complexity $O(n)$ and can be constructed from $\mathcal{M}$ in $O(n)$ time assuming any standard representation of $\mathcal{M}$. There is a natural projection $\pi : \mathcal{M}_o \to \mathcal{M}$ that sends a point in a face f' or f'' of $\mathcal{M}_o$ to the same point in the original face f of $\mathcal{M}$.

There are two boundary cycles σ_1, σ_2 in $\mathcal{M}_o$ such that $\sigma = \pi \circ \sigma_1 = \pi \circ \sigma_2$. A cycle α in $\mathcal{M}$ is homotopic to the boundary σ if and only if there is a cycle α_1 in $\mathcal{M}_o$ homotopic to the boundary σ_1 that satisfies $\alpha = \pi \circ \alpha_1$. Therefore, it holds that if β is a tight cycle homotopic to σ_1 in $\mathcal{M}_o$, then $\pi \circ \beta$ is a tight cycle homotopic to the boundary σ in $\mathcal{M}$. Thus, the problem reduces to finding a tight cycle homotopic to the boundary σ_1 in the orientable combinatorial surface $\mathcal{M}_o$, which we have already solved before. □

This result also implies that we can compute a shortest contractible cycle that encloses a given face f of $\mathcal{M}$ in $O(n \log n)$ time: cut the interior of f out from $\mathcal{M}$ and compute the shortest cycle homotopic to the new boundary.

We next discuss the problem of finding a shortest essentially simple, contractible cycle in a combinatorial surface. However we want to avoid trivial solutions consisting of a single vertex, a walk through a single edge in both directions, or more generally a walk contained in a tree. Thus we require that the part enclosed by the cycle contains at least one face of the graph. To formalize this, let us say that a cycle γ in a combinatorial surface $\mathcal{M}$ is an *enclosing cycle* if it is essentially simple, contractible, and any topological disk in $\mathcal{M} \text{✂} \gamma$ contains a non-empty set of faces. In general $\mathcal{M} \text{✂} \gamma$ has one topological disk, unless $\mathcal{M}$ is a sphere. When $\mathcal{M}$ is not a sphere, a shortest enclosing cycle can be found by finding, for each face f, the shortest contractible cycle that encloses f, and reporting the shortest among them. We next give a faster algorithm that uses a more global approach. Our algorithm reduces the problem to that of finding a shortest enclosing cycle in a topological disk, which equivalent to finding a minimum cut in the dual weighted graph.

THEOREM 5.4. *Let $\mathcal{M}$ be an orientable combinatorial surface with complexity n. We can find in $O(T_{min-cut}(n) + n \log n)$ time a shortest enclosing cycle, where $T_{min-cut}(n)$ is the time needed to find a minimum cut in a planar weighted graph of size n.*

PROOF. If $\mathcal{M}$ is a combinatorial surface homeomorphic to a sphere, the edges in a shortest enclosing cycle correspond to a minimum cut in the weighted graph dual to $G(\mathcal{M})$, and therefore the result follows. If $\mathcal{M}$ is a topological disk, we glue a disk along the boundary of $\mathcal{M}$ to obtain a combinatorial sphere $\mathcal{M}'$. A shortest enclosing cycle in $\mathcal{M}'$ corresponds to a shortest enclosing cycle in $\mathcal{M}$, and therefore the result follows.

It remains the case when $\mathcal{M}$ is not a combinatorial sphere or disk. We first describe the algorithm and then discuss its correctness. The algorithm is as follows. First, compute a tight, essentially simple, non-separating cycle α in $\mathcal{M}$ using Theorem 4.3, and construct the surface $\mathcal{M}_1 = \mathcal{M} \text{✂} \alpha$, which has at least two boundary components. Let b be the number of boundary components of $\mathcal{M}_1$. Then, take $\mathcal{M}_2 = \mathcal{M}_1 \text{✂} (\beta_1, \ldots, \beta_{b-1})$, where $\beta_1, \ldots, \beta_{b-1}$ is the tight system of disjoint arcs arising from Lemma 3.5. It follows that $\mathcal{M}_2$ has precisely one boundary cycle. Next, construct a tight system of disjoint arcs $\alpha_1, \ldots, \alpha_k$ for $\mathcal{M}_2$ as described in Lemma 3.4 and construct the topological disk $\mathcal{M}_3 = \mathcal{M}_2 \text{✂} (\alpha_1, \ldots, \alpha_k)$. Finally compute a shortest enclosing cycle in the disk $\mathcal{M}_3$, and return it as answer. Any surface constructed during the algorithm has complexity $O(n)$, and therefore the procedure we have described takes $O(n \log n) + T_{min-cut}(O(n))$ time.

We next show the correctness of the algorithm. Consider a shortest enclosing cycle γ bounding a disk D_γ that does not contain any other shortest enclosing cycle. The exchange argument used in the proof of Lemma 3.3 shows that a tight cycle or arc α in $\mathcal{M}$ is disjoint from the interior of D_γ. It follows that, for any tight cycle or arc α, a shortest enclosing cycle γ in $\mathcal{M} \text{✂} \alpha$ is also a shortest enclosing cycle in $\mathcal{M}$. Since during the algorithm we only cut along tight arcs and cycles, it is clear that the shortest enclosing cycle in $\mathcal{M}_3$ is a shortest enclosing cycle in the original surface $\mathcal{M}$. □

Currently, the best algorithm for computing minimum cuts in planar graphs takes $O(n \log^2 n)$ time [Chalermsook et al. 2004]. Therefore, we can find a shortest enclosing cycle in a combinatorial surface in $O(n \log^2 n)$ time.

We next consider the problem of computing the girth of an abstract weighted graph, defined as the length of a shortest closed walk without repeating vertices. Note that in the following result we are not assuming any embedding of the input

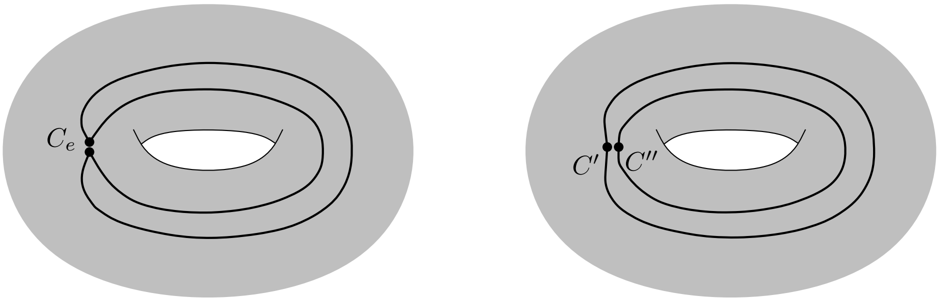

Fig. 6. Figure for the second case of Theorem 5.5. The dots represent the same vertex. A shortest enclosing cycle C_e (left) that repeats some vertex can be split into two cycles C' and C'' that are non-contractible (right).

graph.

THEOREM 5.5. *Let G be a weighted graph with m edges. The girth of G can be found in $O(T_{nc}(m) + m \log^2 m)$ time, where $T_{nc}(n)$ denotes the time needed to find a shortest non-contractible cycle in combinatorial surface with complexity n and genus n.*

PROOF. Note that any graph G with m edges can easily be embedded in an orientable surface of genus $g = O(m)$ [Mohar and Thomassen 2001]. Consider this embedded graph as an orientable combinatorial surface $\mathcal{M}$ without boundary, and compute a shortest non-contractible cycle C_{nc} in $\mathcal{M}$ using $O(T_{nc}(m)$ time and a shortest enclosing cycle C_e in $\mathcal{M}$ using $O(m \log^2 m)$ time. It is known that the cycle C_{nc} does not repeat any vertex of G because of the 3-path property [Cabello and Mohar 2007]. We then distinguish two cases:

— If C_e does not repeat any vertex as well, we return the shortest between C_{nc}, C_e, and the result is clearly correct.

— If C_e does repeat some vertex, then C_{nc} defines the girth of G, and we can just return C_{nc}. To see that indeed C_{nc} defines the girth of G, split the cycle C_e into two cycles C' and C'' at a vertex where C_e passes twice. See Figure 6. It cannot be that both C', C'' are contractible, as otherwise one of them would be enclosing and shorter. Therefore, C' or C'' is non-contractible, and is shorter that C_e. This means C_{nc} is shorter than C_e, and therefore shorter than any enclosing cycle.

□

An algorithm finding shortest non-contractible cycles in a combinatorial surface with complexity n in $O(n^{2-\varepsilon})$ time, for some constant $\varepsilon > 0$, would imply that the girth of a graph with m vertices can be computed in $O(m^{2-\varepsilon})$ time. However, for sparse graphs no algorithms to compute the girth in $O(m^{2-\varepsilon})$ time are currently known; see [Alon et al. 1997] for the best known bounds for unweighted graphs of bounded girth. Therefore, we cannot expect a substantial improvement over the near-quadratic algorithm of [Erickson and Har-Peled 2004] for finding shortest non-contractible cycles, unless the girth of sparse graphs can be computed substantially faster.

REFERENCES

Alon, N., Yuster, R., and Zwick, U. 1997. Finding and counting given length cycles. *Algorithmica 17,* 3, 209–223.

Cabello, S. and Chambers, E. W. 2007. Multiple source shortest paths in a genus g graph. In *SODA '07: Proc. 18th Symp. Discrete Algorithms*. 89–97.

Cabello, S. and Mohar, B. 2007. Finding shortest non-separating and non-contractible cycles for topologically embedded graphs. *Discrete Comput. Geom. 37*, 213–235. Preliminary version in ESA '05.

Chalermsook, P., Fakcharoenphol, J., and Nanongkai, D. 2004. A deterministic near-linear time algorithm for finding minimum cuts in planar graphs. In *SODA '04: Proc. 15th Symp. Discrete Algorithms*. 828–829.

Chambers, E. W., Colin de Verdière, É., Erickson, J., Lazarus, F., and Whittlesey, K. 2006. Splitting (complicated) surfaces is hard. In *SOCG '06: Proc. 22nd Symp. Comput. Geom.* 421–429.

Colin de Verdière, É. and Erickson, J. 2006. Tightening non-simple paths and cycles on surfaces. In *SODA '06: Proc. 17th Symp. Discrete Algorithms*. 192–201.

É. Colin de Verdière and F. Lazarus. 2004. Optimal pants decompositions and shortest homotopic cycles on an orientable surface. In *11th International Symposium on Graph Drawing, Perugia, Italy*. LNCS, vol. 2912. Springer, 478–490.

É. Colin de Verdière and F. Lazarus. 2005. Optimal system of loops on an orientable surface. *Discrete Comput. Geom. 33*, 507–534. Preliminary version in FOCS '02.

Eppstein, D. 2003. Dynamic generators of topologically embedded graphs. In *SODA '03: Proc. 14th Symp. Discrete Algorithms*. 599–608.

Erickson, J. and Har-Peled, S. 2004. Optimally cutting a surface into a disk. *Discrete Comput. Geom. 31*, 37–59. Preliminary version in SOCG '02.

Erickson, J. and Whittlesey, K. 2005. Greedy optimal homotopy and homology generators. In *SODA '03: Proc. 16th Symp. Discrete Algorithms*. 1038–1046.

Frederickson, G. N. 1987. Fast algorithms for shortest paths in planar graphs, with applications. *SIAM J. Comput. 16,* 6, 1004–1004.

Hass, H. and Scott, P. 1985. Intersections of curves on surfaces. *Israel J. Math. 51*, 90–120.

Hatcher, A. 2001. *Algebraic Topology*. Cambridge University Press. Available at `http://www.math.cornell.edu/~hatcher/`.

Kawarabayashi, K. and Reed, B. 2007. Computing crossing number in linear time. In *STOC '07: Proc. 39th ACM Symp. Theory Comput.* 382–390.

Kutz, M. 2006. Computing shortest non-trivial cycles on orientable surfaces of bounded genus in almost linear time. In *SOCG '06: Proc. 22nd Symp. Comput. Geom.* 430–438.

Massey, W. S. 1967. *Algebraic Topology: An Introduction*. Springer Verlag.

Mohar, B. and Thomassen, C. 2001. *Graphs on Surfaces*. Johns Hopkins University Press, Baltimore.

Stillwell, J. 1993. *Classical Topology and Combinatorial Group Theory*. Springer-Verlag, New York.