Transforming Curves on Surfaces Redux*

Jeff Erickson† Kim Whittlesey

University of Illinois at Urbana-Champaign

{jeffe,kwhittle}@illinois.edu

Abstract

Almost exactly 100 years ago, Max Dehn described the first combinatorial algorithm to determine whether two given cycles on a compact surface are homotopic, meaning one cycle can be continuously deformed into the other without leaving the surface. We describe a simple variant of Dehn's algorithm that runs in linear time, with no hidden dependence on the genus of the surface. Specifically, given two closed vertex-edge walks of length at most ℓ in a combinatorial surface of complexity n, our algorithm determines whether the walks are homotopic in $O(n+\ell)$ time. Our algorithm simplifies and corrects a similar algorithm of Dey and Guha [*JCSS* 1999] and simplifies the more recent algorithm of Lazarus and Rivaud [FOCS 2012], who identified a subtle flaw in Dey and Guha's results. Our algorithm combines components of these earlier algorithms, classical results in small cancellation theory by Gersten and Short [*Inventiones* 1990], and simple run-length encoding.

1 Introduction

Almost exactly 100 years ago, Max Dehn [8] described the first combinatorial algorithms for two fundamental topological problems involving closed curves on compact surfaces: determining whether a given closed curve can be continuously deformed to a point, and determining whether one given closed curve can be continuously deformed into another. In modern terminology, Dehn's algorithms determine whether a given cycle is *contractible*, and whether two given cycles are *freely homotopic*. Both of these problems were already known to be solvable using covering space techniques, originally due to Schwarz [24, 38] and developed further by Poincaré [33] and Dehn himself [7]. The key insight in Dehn's 1912 paper [8] is that any contractible cycle can be contracted by a sequence of greedy local moves, each involving only a small segment of the cycle; moreover, these local moves can be performed directly on the cycle without constructing or searching any portion of any covering space. Dehn's algorithm exposed deep connections between fundamental groups of surfaces and the geometry of the hyperbolic plane, which laid the foundation for several fields of mathematics, most notably combinatorial and geometric group theory and the study of non-positively curved metric spaces [5, 14, 18, 19, 28, 31, 32, 39]. In particular, Dehn's greedy algorithm has been broadly generalized to a family of techniques called "small cancellation theory" [14, 17, 19, 28, 29].

For any *fixed* surface, careful implementations of Dehn's algorithms run in time linear in the complexity of the input curve [3, 11, 15, 21]; however, this "linear" running time hides a constant that depends on the genus of the surface. We consider a formulation where the surface is not fixed, but rather provided as part of the input. Specifically, the input consists of an arbitrary polygonal decomposition of the surface with complexity n and one or two closed walks with length at most ℓ in the 1-skeleton of this decomposition. We analyze our algorithms in the standard integer RAM model.

In this setting, Schipper [36] solved the contractibility problem in $O(gn + g^2\ell)$ time, where g is the genus of the input surface, by constructing and searching a small relevant portion of the universal cover of the input surface, as originally suggested by Schwarz. Dey and Schipper [10] later improve the running time to $O(n + \ell \log g)$. Dey and Guha [9] adapted Dehn's "small cancellation" contractibility algorithm to run in optimal $O(n + \ell)$ time. (We describe some of the subtleties in Dey and Guha's algorithm in Section 3.) Dey and Guha also claimed a linear-time algorithm to test whether two cycles are freely homotopic; however, Lazarus and Rivaud [26] recently discovered a subtle flaw in their algorithm. Lazarus and Rivaud described a different $O(n + \ell)$-time homotopy-testing algorithm that constructs and searches a finite portion of a certain *cyclic* covering space. Their algorithm adapts and simplifies techniques of Colin de Verdière and Erickson to find the shortest cycle in a given free homotopy class [6].

1.1 Our Results

This paper recasts Lazarus and Rivaud's homotopy-testing algorithm into the language of small cancellation theory, thereby avoiding the need to construct or search any portion of any covering space. In fact, aside from the input surface itself, the most complicated data structure used by our algorithm is a circular linked list.

*Portions of this work were done while the authors were visiting IST Austria. See http://www.cs.uiuc.edu/~jeffe/pubs/dehn.html for the most recent version of this paper.

†This author's work was partially supported by NSF grant CCF 09-15519.

Following Lazarus and Rivaud, we first reduce our problems in arbitrary surface complexes to the same problems in a particular complex we call a *system of quads*. A system of quads on an orientable surface of genus $g > 0$ is a cell complex with two vertices, $4g$ edges between those vertices, and $2g$ quadrilateral faces. Unlike Dey and Guha's reduction to *systems of loops*, our reduction to systems of quads increases the complexity of the input walk by at most a factor of 2. We describe this reduction in Section 3.

Next, we encode each input walk as a cyclic sequence of *turns*; each turn is the number of corners (modulo $4g$) in clockwise order between two successive edges in the cycle. For example, a turn of 0 represents a *spur* (a directed edge followed by its reversal), a turn of 1 represents a sharp left turn, and a turn of -1 represents a sharp right turn. To speed up computation, we compress these turn sequences using simple run-length encoding.

Small cancellation results of Gersten and Short [17], which we prove in Section 4 using a combinatorial version of the Gauss-Bonnet theorem, imply that any nontrivial contractible cycle contains either a spur or a *bracket* (a path around three sides of a rectangle). More generally, we define a cycle to be *reduced* if it has no spurs or brackets. Gersten and Short's results also imply that a cycle in a system of quads is reduced if and only if there is no shorter cycle in the same free homotopy class. We can easily remove any spur or bracket from the run-length-encoded turn sequence in constant time without changing the homotopy type of the encoded cycle. We describe how to remove all spurs and brackets from a run-length-encoded turn sequence in linear time using a variant of Dehn's algorithm. We immediately obtain a linear-time contractibility algorithm.

Finally, in Section 5, we define a cycle to be *canonical* if it contains no sharp right turns (-1) and not all its turns are equal to -2. We show that any reduced cycle can be transformed into a canonical cycle in the same free homotopy class, using a single scan of run-length-encoded turn sequence. Intuitively, this transformation pushes the reduced cycle as far to the right as possible without increasing its length. Finally, using the combinatorial Gauss-Bonnet theorem again, we prove that there is exactly one canonical cycle in each free homotopy class. Our algorithm reports that two input cycles are freely homotopic if and only if they are transformed into the same canonical cycle.

1.2 Simplifying Assumptions

We explcitly consider only *orientable* combinatorial surfaces *without boundary* and with genus at least 2, but our results can be extended to arbitrary combinatorial surfaces with little difficulty. As Lazarus and Rivaud observe [26], non-orientable surfaces can be handled by passing to the oriented double cover. All cycles are contractible on the sphere (genus 0), and it is easy to determine whether two cycles are homotopic on the torus (genus 1) using homology. Finally, homotopy problems on surfaces with boundary can be solved quickly by even simpler linear-time algorithms.

2 Background

We begin by recalling several standard definitions from combinatorial surface topology and topological graph theory. For more detailed background, we refer the reader to Stillwell [39], Mohar and Thomassen [30], and Edelsbrunner and Harer [12].

2.1 Combinatorial Surfaces

A ***2-manifold*** is a Hausdorff space in which every point has an open neighborhood homeomorphic to the plane $\mathbb{R}^2$. A 2-manifold is ***non-orientable*** if it has a subset homeomorphic to a Möbius band and ***orientable*** otherwise. We consider only compact, connected, orientable 2-manifolds in this paper.

A ***combinatorial surface*** Σ is a decomposition of a compact 2-manifold *without boundary* into ***vertices***, ***edges***, and ***faces***; every vertex is a point, every edge is a simple path between vertices, and every face is an open disk. The underlying graph of vertices and edges is called the ***1-skeleton*** of Σ. Each incidence between a face and a vertex of Σ is called a ***corner***; each incidence between a vertex and an edge is called a ***dart***. Any combinatorial surface Σ can be represented by a ***rotation system***, which records the clockwise cyclic order of darts at each vertex.

The ***dual*** of a combinatorial surface Σ is another combinatorial surface Σ^* on the same 2-manifold, with a vertex f^* for each face f of Σ, an edge e^* for each edge e of Σ, and a face v^* for each vertex v of Σ. A ***spanning tree*** of a combinatorial surface Σ is a spanning tree of its 1-skeleton; a ***spanning cotree*** of Σ is a set of edges whose duals form a spanning tree of Σ^*. A ***tree-cotree decomposition*** of Σ is a partition (T, L, C) of its edges into three edge-disjoint subgraphs: a spanning tree T, a spanning cotree C, and the leftover edges L [2, 13].

A ***combinatorial surface with boundary*** is obtained from a combinatorial surface by deleting a subset of its faces. The ***boundary*** of such a surface is the set of facial walks of the deleted faces; the boundary is ***non-singular*** if it consists of disjoint simple cycles and ***singular*** otherwise. We emphasize that a combinatorial surface with *singular* boundary is homotopy-equivalent but *not* homeomorphic to a 2-manifold with boundary; a combinatorial surface with *non-singular* boundary is actually a 2-manifold with boundary. An ***interior*** vertex is a vertex that does not lie on any boundary walk.

2.2 Paths, Cycles, and Homotopy

A ***path*** in an abstract 2-manifold Σ is a continuous function $\pi\colon [0,1] \to \Sigma$; a ***cycle*** is a continuous function $\gamma\colon S^1 \to \Sigma$. A path π whose endpoints coincide is called a ***loop***; the common endpoint $\pi(0) = \pi(1)$ is called the ***basepoint*** of the loop. A path or cycle is ***simple*** if it is injective; a loop is simple if it injective except at the basepoint. Whenever we speak of a path or cycle in a *combinatorial* surface, we always mean a walk or circuit in its 1-skeleton.

The ***concatenation*** $\boldsymbol{\pi \cdot \pi'}$ of two paths π and π' with $\pi(1) = \pi'(0)$ is the path with $(\pi \cdot \pi')(t) = \pi(2t)$ for all $t \le 1/2$ and $(\pi \cdot \pi')(t) = \pi'(2t-1)$ for all $t \ge 1/2$. The ***reversal*** $\boldsymbol{\overline{\pi}}$ of π satisfies $\overline{\pi}(t) = \pi(1-t)$ for all t; the reversal of a cycle is defined similarly.

The ***genus*** $\boldsymbol{g}$ of a surface Σ is the maximum possible number of disjoint cycles $\gamma_1, \dots, \gamma_g$ whose complement $\Sigma \setminus (\gamma_1 \cup \cdots \cup \gamma_g)$ is connected. For example, the sphere and the disk have genus 0, the torus and the projective plane have genus 1, and the Klein bottle has genus 2.

A ***path homotopy*** from a path π to another path π' is a function $h\colon [0,1]^2 \to \Sigma$ such that $h(0,t) = \pi(t)$ and $h(1,t) = \pi'(t)$ for all t, and $h(s,0) = \pi(0) = \pi'(0)$ and $h(s,1) = \pi(0) = \pi'(0)$ for all s. Two paths are *homotopic* if there is a path homotopy from one to the other. A ***free homotopy*** between two cycles γ and γ' is a function $h\colon [0,1] \times S^1 \to \Sigma$ such that $h(0,\theta) = \gamma(\theta)$ and $h(1,\theta) = \gamma'(\theta)$ for all $\theta \in S^1$. Two cycles are ***freely homotopic*** if there is a free homotopy from one to the other. A cycle γ is ***contractible*** if it is freely homotopic to a constant cycle. Equivalently, two paths π and π' are homotopic if and only if the loop $\pi \cdot \overline{\pi}'$ is contractible.

2.3 Euler and Gauss-Bonnet

Let Σ be a combinatorial surface, possibly with singular boundary. The ***Euler characteristic*** $\boldsymbol{\chi(\Sigma)}$ of Σ is the number of vertices and faces minus the number of edges. Euler's formula states that $\chi(\Sigma) = 2 - 2g - b$, where g is the genus of Σ and b is the number of boundary cycles.

Suppose we assign an arbitrary real number $\angle c$ to each corner c of Σ, called the *exterior angle* at c. We define the ***(combinatorial) curvature*** of a face f or a vertex v as follows:

$$\kappa(f) := 1 - \sum_{c \in f} \angle c,$$

$$\kappa(v) := 1 - \frac{1}{2}\deg(v) + \sum_{c \in v} \angle c.$$

Here $\deg(v)$ denotes the number of darts in Σ that are incident to v. To simplify notation, we identify each face or vertex with its set of incident corners, and we measure angles in *circles* instead of radians or degrees.

The Combinatorial Gauss-Bonnet Theorem. *For any combinatorial surface $\Sigma = (V, E, F)$, possibly with singular boundary, and for any assignment of angles to the corners of Σ, we have $\sum_{f \in F} \kappa(f) + \sum_{v \in V} \kappa(v) = \chi(\Sigma)$.*

Proof: We immediately have $\sum_f \kappa(f) = |F| - \sum_c \angle c$ and $\sum_v \kappa(v) = |V| - |E| + \sum_c \angle c$, which implies that $\sum_f \kappa(f) + \sum_v \kappa(v) = |V| - |E| + |F| = \chi(\Sigma)$. □

In all our applications of this theorem, we assign $\angle c = 1/4$ at every corner c; we emphasize, however, that the theorem holds for *arbitrary* angle assignments. Variants and special cases of this theorem were proposed independently by many different authors [1, 17, 19, 20, 22, 27, 28, 35]; our general formulation is closest to that of McCammond and Wise [29].

2.4 Diagrams

Unlike many previous authors [6, 9, 10, 26, 36], who formulated their results in terms of covering spaces, we rely on a more general class of complexes called ***diagrams***, which are central tools in combinatorial and geometric group theory. Our definitions closely follow McCammond and Wise [29].

Let Δ and Σ be combinatorial surfaces, possibly with singular boundary. We call Δ a ***diagram over*** $\boldsymbol{\Sigma}$ if there is a continuous function $\delta\colon \Delta \to \Sigma$ that sends vertices to vertices, edges to edges, and faces to faces; the function δ is called a ***diagram map***. A diagram is ***singular*** if its boundary is singular, and ***non-singular*** otherwise.

Consider two faces f and f' in a diagram Δ whose boundaries intersect in a path from vertex u to vertex v; the boundary of $f \cup f'$ consists of two paths π and π' from u to v. We say that f and f' ***cancel*** if the diagram map δ sends π and π' to the same path in Σ. Equivalently (because Σ is an orientable surface), two faces in Δ cancel if they share an edge and the diagram map sends them to the same face of Σ but with opposite orientations. A diagram is ***reduced*** if no two of its faces cancel. Any interior vertex of a reduced diagram has the same local neighborhood structure as the corresponding vertex in the target surface Σ. Any subcomplex of a covering space of Σ is also a reduced diagram over Σ, but not every reduced diagram is a subcomplex of a covering space.

A contractible diagram is called a ***disk diagram*** (or a *van Kampen diagram*, or a *Dehn diagram*). Similarly, a diagram with the homotopy type of a circle is called an ***annular diagram*** (or a *conjugacy diagram*, or a *Schupp diagram*). Figure 1 shows a singular annular diagram.

Our algorithms rely on the following fundamental lemmas; for proofs, we refer the reader to Lyndon and Schupp [28, Chapter V] or McCammond and Wise [29, Lemmas 2.17 and 2.18].

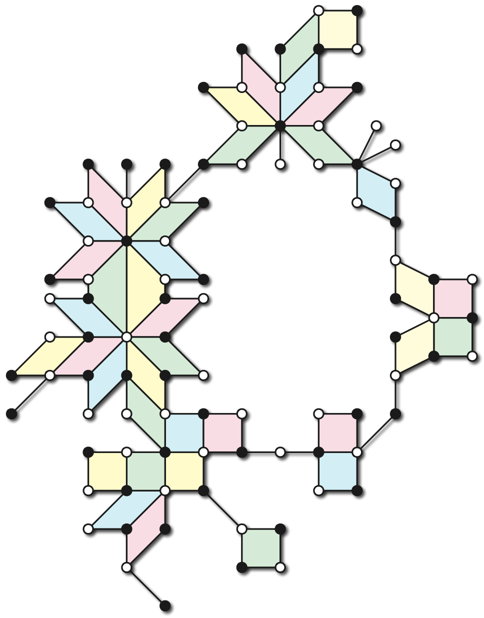

Figure 1. A singular reduced annular diagram over the system of quads in Figures 2(c) and 3; vertex and face colors describe the diagram map.

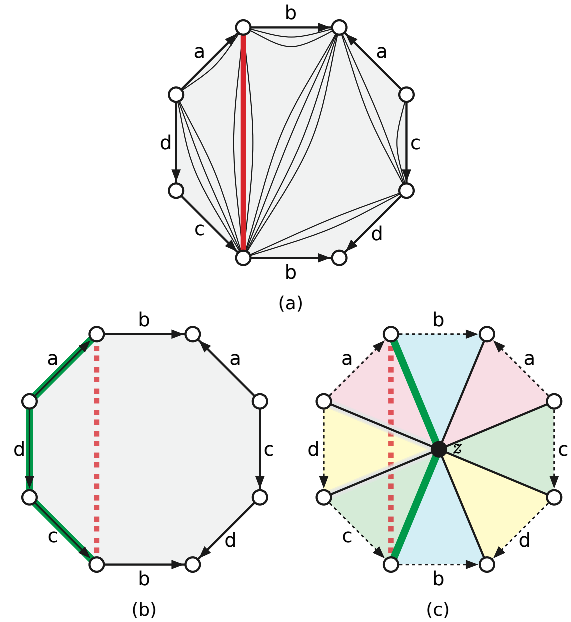

Figure 2. (a) A fundamental polygon obtained by contracting a spanning tree and cutting a system of loops. (b) Replacing a non-tree edge with a walk in the system of loops. (c) Replacing a non-tree edge with a path of length 2 in the corresponding system of quads; see Figure 3.

Lemma 2.1 (Van Kampen [23], Lyndon [27]). *A cycle α in Σ is contractible if and only if there is a reduced disk diagram Δ_α whose boundary walk is sent to α by the diagram map.*

Lemma 2.2 (Schupp [37]). *Two cycles α and β in Σ are freely homotopic if and only if there is a reduced annular diagram $\Delta_{\alpha,\beta}$ whose boundary walks are sent to α and β by the diagram map.*

3 Systems of Quads

Most descriptions of Dehn's algorithms assume *a priori* that the input surface has a single vertex and a single face; such a combinatorial surface is often called a ***system of loops***. Dey and Schipper [10] and Dey and Guha [9] describe preprocessing algorithms that efficiently reduce homotopy problems on arbitrary combinatorial surfaces to the special case of systems of loops.

Dey and Guha's reduction can be described as follows. Let (T, L, C) be an arbitrary tree-cotree decomposition of the input surface Σ. Contracting every edge in T and deleting every edge in C transforms Σ into a system of loops Λ. Each edge in L survives as a loop in Λ; let a denote the basepoint (or "apex") of these loops. Cutting the surface along every loop in L yields a ***fundamental polygon*** P with $2g$ vertices and edges. If we merely contract T and cut along L, each edge in C survives as a path between two corners of this polygon.

For each directed edge e in Σ, we define a walk $\lambda(e)$ in Λ as follows. If $e \in T$, then $\lambda(e)$ is the empty walk; otherwise, $\lambda(e)$ is one of the two walks around the boundary of P with the same endpoints as e. For any cycle α in Σ, we define a cycle $\lambda(\alpha)$ by concatenation. Two cycles α and β in Σ are freely homotopic if and only if the corresponding cycles $\lambda(\alpha)$ and $\lambda(\beta)$ in Λ are also freely homotopic; in particular, α is contractible if and only if $\lambda(\alpha)$ is contractible.

However, the cycle $\lambda(\alpha)$ could be a factor of $\Omega(g)$ longer than α. Dey and Guha hide this increase in length by recording only the first and last segments of each segment $\lambda(e)$. Efficiently maintaining this compact representation complicates their algorithm considerably.

Following Lazarus and Rivaud [26], we avoid this complication by reducing the input surface to a different standard form, which we call a ***system of quads***. A system of quads Q is obtained from a system of loops Λ by introducing a new vertex z (for "zenith" or "Zentrum") in the interior of the unique face of Λ, adding edges between z and every corner of that face, and then deleting the edges of Λ. The resulting combinatorial surface has exactly $4g$ edges (each with endpoints a and z) and $2g$ quadrilateral faces (one for each loop in Λ).

For each directed edge e in Σ, we define a path $q(e)$ in Q as follows. If $e \in T$, then $q(e)$ is the empty path; otherwise, consider e an edge between two corners of P, and define $q(e)$ to be the path of length 2 in Q from one corner to z and then to the other corner. For each edge $e \in L$, there are two possibilities for the path $q(e)$; we choose one arbitrarily. Finally, for any cycle α in Σ, we define the corresponding cycle $q(\alpha)$ in Q by concatenation. Again, it is easy to verify that two cycles α and β are freely homotopic if and only if $q(\alpha)$ and $q(\beta)$ are freely

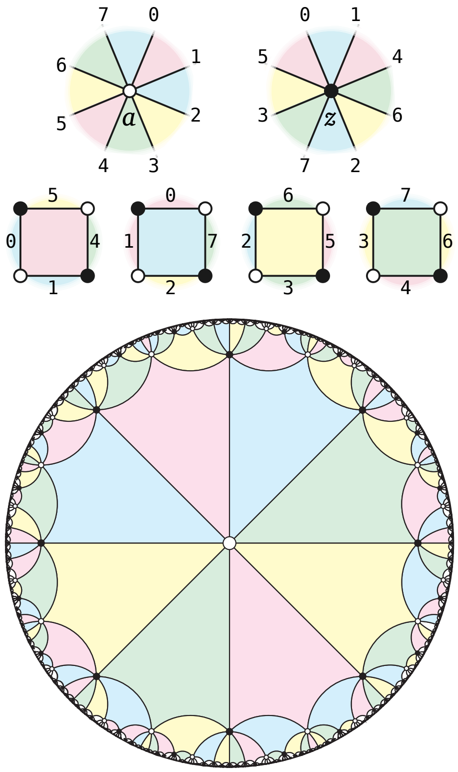

Figure 3. Vertex neighborhoods, faces, and the universal cover of a genus-2 system of quads. Compare with Figures 1 and 2(c).

homotopic. Moreover, the length of $q(\alpha)$ is clearly at most twice the length of α.

Given any combinatorial surface Σ with complexity n and positive genus, it is straightforward to compute a system of quads Q and the paths $q(e)$ for every edge e in Σ in $O(n)$ time. Then for any walk α in Σ with ℓ edges, we can easily compute the corresponding walk $q(\alpha)$ in $O(\ell)$ time. **In light of this reduction, for the rest of the paper we assume without loss of generality that the input surface map Σ is already a system of quads.**

The rest of our algorithm implicitly relies on the fact that a system of quads for a orientable surface of genus two or higher is *hyperbolic* in the sense of Gromov [17, 19]. The universal cover of a system of quads is isomorphic to a regular tiling of the hyperbolic plane by squares meeting at vertices of degree at least 8; see Figure 3 for an example. More importantly, any interior vertex in a reduced diagram over a system of quads has degree at least 8.

4 Contractibility: Reduced Cycles

We now describe our linear-time algorithm to test whether a given cycle α in a system of quads is contractible. Our algorithm greedily shortens α, without changing its homotopy class, by replacing certain subpaths called *spurs* and *brackets* that are obviously not as short as possible with their corresponding shortest paths. A lemma of Gersten and Short [17], which we prove using the combinatorial Gauss-Bonnet theorem, implies that *every* nontrivial contractible cycle contains either a spur or a bracket. Thus, removing all spurs and brackets from a contractible cycle eventually reduces it to the empty cycle. To identify spurs and brackets quickly, we represent the input α as a run-length encoded sequence of *turns*.

4.1 Turn Sequences

Formally, the input cycle α is given as an alternating cyclic sequence $(v_0, e_1, v_1, e_2, \dots, e_\ell)$ of vertices and edges, where for each index i, vertices v_{i-1} and $v_{i \bmod \ell}$ are the endpoints of edge e_i.[1] For any two edges e and e' that share an endpoint v, we define the ***turn*** $\boldsymbol{\tau(e, v, e')}$ to be the number of corners between e and e' in clockwise order around v_i. The ***turn sequence*** $\boldsymbol{\tau(\alpha)}$ is the cyclic sequence $(\tau_0, \tau_1, \dots, \tau_\ell)$, where $\tau_i = \tau(e_i, v_i, e_{i+1 \bmod \ell})$ for all i. Computing this turn sequence in $O(\ell)$ time is straightforward. For notational convenience, we write $\bar{t} = (-t) \bmod 4g$ for any integer t.

To simplify exposition, we use exponents to denote a run of identical turns in the turn sequence; for example, the expression $(1, 2^4, 1^2, 2)$ represents the turn sequence $(1, 2, 2, 2, 2, 1, 1, 2)$. Our algorithm uses a similar compression scheme called ***run-length encoding***. Instead of the raw turn sequence, we actually store a sequence of pairs of the form $((\tau_0, r_0), (\tau_1, r_1), \dots (\tau_{k-1}, r_{k-1}))$, with $\tau_i \neq \tau_{i+1 \bmod k}$ for all i, representing the turn sequence $(\tau_0^{r_0}, \tau_1^{r_1}, \dots, \tau_{k-1}^{r_{k-1}})$. Again, it is straightforward to compute the run-length-encoded turn sequence of any cycle α in $O(\ell)$ time.

4.2 Spurs and Brackets

A ***spur*** in a cycle α is any vertex with turn 0. A ***left bracket*** is a subpath of α where each endpoint has a turn 1 and every other vertex has turn 2. Similarly, a ***right bracket*** is a subpath of α where each endpoint has turn $\bar{1}$ and every other vertex has turn $\bar{2}$.

Lemma 4.1 (Gersten and Short [17, Corollary 5.1]). *The boundary of any non-singular reduced disk diagram over a system of quads has either four left brackets or four right brackets.*

Proof: Let Δ be a non-singular reduced diagram over a system of quads Σ. Because Δ is non-singular, it has at least one face, and therefore at least four edges. Orient the boundary of Δ counterclockwise, so that the turn at any boundary vertex is equal to the number of faces of Δ incident to that vertex. Call a boundary vertex of Δ *convex*

[1] Because α is a cycle in a system of quads, ℓ must be even, $v_i = a$ for all even i, and $v_i = z$ for all odd i, but we never actually use these facts.

if its turn is 1, *flat* if its turn is 2, and *concave* otherwise. Assign an angle of $1/4$ to every corner of Δ. Then every face has curvature zero; every convex boundary vertex has curvature $1/4$; every flat boundary vertex has curvature 0; and every other vertex has curvature at most $-1/4$. The combinatorial Gauss-Bonnet theorem gives us $\sum_v \kappa(v) = 1$. Thus, Δ has at least four more convex boundary vertices than concave boundary vertices. It follows immediately that the boundary of Δ has at least four right brackets. Symmetrically, if the boundary of Δ is oriented clockwise, it has at least four right brackets. □

Corollary 4.2. *Every nontrivial contractible cycle in a system of quads has either a spur or four brackets.*

Proof: Let γ be a non-empty contractible loop in Σ. Van Kampen's Lemma 2.1 implies that there is a reduced disk diagram Δ whose boundary is mapped to γ. If Δ is non-singular, then γ and the boundary of Δ have identical turn sequences, so Lemma 4.1 implies that γ has at least four brackets. If any vertex of Δ has degree 1, that vertex must map to a spur in γ.

Otherwise, Δ is singular but has no vertices of degree 1. In this case, Δ consists of a "tree" of non-singular reduced disk diagrams that are connected by paths and cut vertices. At least two of these non-singular sub-diagrams Δ_1 and Δ_2 have only one vertex in common with the rest of Δ. Lemma 4.1 implies that the boundary of Δ_1 has at least four brackets. At least two of these brackets do not overlap the cut vertex joining Δ_1 to the rest of Δ; the diagram map sends these two brackets to brackets in γ. Similarly, γ also has at least two brackets from the boundary of Δ_2. □

Finally, we say that a cycle is ***reduced*** if it contains no spurs or brackets. Corollary 4.2 immediately implies that every reduced cycle is either trivial or non-contractible.

4.3 Elementary Reductions

Any spur or bracket can be removed from a cycle by a local modification that does not change its homotopy type. We call such local modifications ***elementary reductions***. Our algorithm actually performs these elementary reductions by modifying the run-length-encoded turn sequence rather than the cycle itself.

We call a spur or bracket ***standard*** if it excludes at least three edges of the cycle, ***near-cyclic*** if it excludes exactly two edges, and ***cyclic*** if it uses every edge of the cycle. It is not possible for a spur or bracket to exclude only one edge of a cycle. A non-trivial cycle in a system of quads has at least two edges, so a spur cannot be cyclic. For each of the eight remaining cases, we have a different elementary reduction—two for spurs, three for left brackets, and three for right brackets. These elementary reductions are illustrated in Figure 4, and their effects on the turn

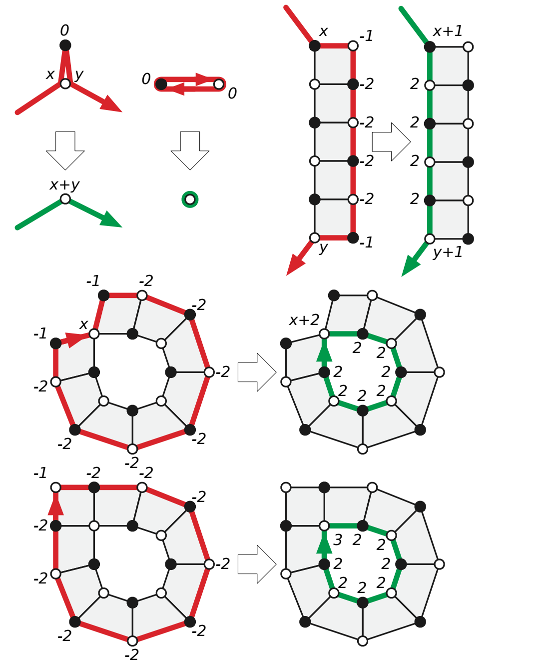

Figure 4. Elementary reductions removing a standard spur, a near-cyclic spur, a standard right bracket, an near-cyclic right bracket, and a cyclic right bracket. Symmetric elementary reductions remove left brackets.

sequence are listed below. We emphasize that reducing a non-standard spur or brackets modifies the *entire* turn sequence, as indicated by the parentheses below.

$$x,\ 0,\ y \rightsquigarrow x+y$$
$$(0,\ 0) \rightsquigarrow (\ )$$

$$x,\ 1,\ 2^r,\ 1,\ y \rightsquigarrow x-1,\ \bar{2}^r,\ y-1$$
$$(x,\ 1,\ 2^r,\ 1) \rightsquigarrow (x-2,\ \bar{2}^r)$$
$$(1,\ 2^r) \rightsquigarrow (\bar{3},\ \bar{2}^{r-2})$$

$$x,\ \bar{1},\ \bar{2}^r,\ \bar{1},\ y \rightsquigarrow x+1,\ 2^k,\ y+1$$
$$(x,\ \bar{1},\ \bar{2}^r,\ \bar{1}) \rightsquigarrow (x+2,\ 2^r)$$
$$(\bar{1},\ \bar{2}^r) \rightsquigarrow (3,\ 2^{r-2})$$

4.4 Reducing in Linear Time

Since each elementary reduction decreases the length of a cycle by 2, any cycle of length ℓ can be transformed into a reduced cycle using at most $\ell/2$ elementary reductions. It remains only to show that we can find and execute these elementary reductions in linear time.

Lemma 4.3. *Given a cycle α of length ℓ in a system of quads, we can compute a reduced cycle freely homotopic to α in $O(\ell)$ time.*

Proof: We begin by computing the run-length-encoded turn sequence of α in $O(\ell)$ time. We store the runs in a circular doubly-linked list, so that each elementary move can be performed in constant time. The turns x and y adjacent to a spur or bracket may lie in nontrivial runs, or even in the same run; nevertheless, each elementary reduction replaces at most five old runs with at most five new runs.

In addition to the value τ_i and run-length r_i of each run of identical turns, we also compute and maintain the directed edge e_i leading into the first turn of the run. These edges allow us to recover the reduced cycle α' from its run-length-encoded turn sequence in $O(\ell)$ time. Maintaining the edges e_i during an elementary reduction requires only $O(1)$ additional time; we omit the tedious details. (Maintaining these edges is not actually necessary when we are only testing contractibility, but our free-homotopy algorithm requires them.)

The main reduction algorithm scans through the run sequence one run at a time, finding and reducing spurs and brackets. Initially, we mark every run *dirty*. We maintain an index i into the run sequence. In each iteration, if runs $i-4$ through i do not contain a spur or a bracket, we mark run $i-4$ *clean* and increment i. (All index arithmetic is performed modulo the current number of runs.) Otherwise, we perform an elementary reduction, replacing those five runs with at most five new runs; we then mark all the modified runs *dirty* and decrease i by 5. The algorithm repeats these iterations until either all runs are marked *clean*, in which case we are done, or at most five runs remain in the run sequence, in which case we can complete the reduction in $O(1)$ additional time. (Lemma 4.2 implies that every bracket in a contractible cycle is standard, so if we encounter a near-cyclic or cyclic bracket, we can immediately report that α is non-contractible.)

To analyze the main algorithm, consider the potential $\Phi = c + 8r$, where c denotes the current number of clean runs and r denotes the number of elementary reductions that have been performed so far. Initially, we have $\Phi = 0$. Because each elementary reduction decreases the length of the encoded cycle by 2, we have $r \le \ell/2$. Similarly, because the number of runs cannot exceed the number of edges, we immediately have $c \le \ell$. Thus, $\Phi \le 5\ell$. Each iteration runs in $O(1)$ time and increases Φ by at least 1. We conclude that the total running time is $O(\ell)$. $\square$

Theorem 4.4. *Given a combinatorial surface Σ with complexity n and a cycle α in Σ with length ℓ, we can determine whether α is contractible in Σ in $O(n+\ell)$ time.*

5 Free Homotopy: Canonical Cycles

Finally, we describe our algorithm to determine whether two given cycles in a system of quads are freely homotopic. Following a common strategy first proposed by Dehn [8], we transform each input cycle into a unique *canonical* cycle in its free homotopy class. Then two cycles are freely homotopic if and only if they are transformed into the same canonical cycle.

An intuitively natural choice for the canonical cycles, also suggested by Dehn [8], are the *minimum-length* cycles in each homotopy class; indeed, these are precisely the cycles computed by our reduction algorithm. Unfortunately, almost all free homotopy classes in a system of quads contain more than one shortest cycle; freely homotopic *reduced* cycles are not necessarily equal. Following Lazarus and Rivaud [26], we intuitively declare a cycle to be canonical if it is the *rightmost* reduced cycle in its homotopy class. (Dey and Guha [9] proposed a similar strategy for their free-homotopy algorithm, but as Lazarus and Rivaud recently observed [26], their proposed canonical cycles were not actually unique.)

5.1 Elementary Right Shifts

More formally, we say that a cycle is ***canonical*** if it is reduced, its turn sequence contains no $\bar{1}$, and its turn sequence does not contain only $\bar{2}$s. A turn of $\bar{1}$ indicates that the cycle can be shifted to the right using an elementary homotopy of the form $x,\ \bar{1},\ y \rightsquigarrow x+1, 1, y+1$. However, this change may introduce new $\bar{1}$s. To avoid cascading, we define an ***right bend*** to be a subpath with turn sequence $\bar{2}^s, \bar{1}, \bar{2}^t$ for some integers s and t. Any right bend can be eliminated by shifting the entire path one step to the right; we call each such move an ***elementary right shift***. Exhaustive case analysis (which we omit) implies that there are seven types of elementary right shift, differentiated by which of the integers s and t are positive, whether the right bend covers the entire cycle, and how the right bend meets itself if it does cover the whole cycle. We also consider the global substitution $(\bar{2}^\ell) \rightsquigarrow (2^\ell)$ to be an eighth type of elementary right shift.

The different types of elementary right shift are listed below and illustrated in Figure 5; again, parentheses in the last five cases indicate that we are modifying the entire turn sequence. Right shifts marked $(+)$ require the exponents s and t to be positive; right shifts marked $(*)$ require $x \neq \bar{3}$.

$$
\begin{aligned}
x,\ \bar{2}^s,\ \bar{1},\ \bar{2}^t,\ y &\rightsquigarrow x+1,\ 1,\ 2^{s-1},\ 3,\ 2^{t-1},\ 1,\ y+1 && (+)\\
x, \bar{1}, \bar{2}^t,\ y &\rightsquigarrow x+1,\ 2^t,\ 1,\ y+1\\
x, \bar{2}^s,\ \bar{1},\ y &\rightsquigarrow x+1,\ 1,\ 2^s,\ y+1\\
(x,\ \bar{2}^s,\ \bar{1},\ \bar{2}^t) &\rightsquigarrow (z+2,\ 1,\ 2^{s-1},\ 3,\ 2^{t-1},\ 1) && (+)(*)\\
(x,\ \bar{1},\ \bar{2}^t) &\rightsquigarrow (z+2,\ 2^t,\ 1) && (*)\\
(x,\ \bar{2}^s,\ \bar{1}) &\rightsquigarrow (z+2,\ 1,\ 2^s) && (*)\\
(\bar{3},\ \bar{2}^s,\ \bar{1},\ \bar{2}^t) &\rightsquigarrow (1,\ 2^s,\ 3,\ 2^t)\\
(\bar{2}^\ell) &\rightsquigarrow (2^\ell)
\end{aligned}
$$

In all these moves, neither of the turns x and y can be equal to $\bar{1}$ or 0 because we apply elementary right shifts

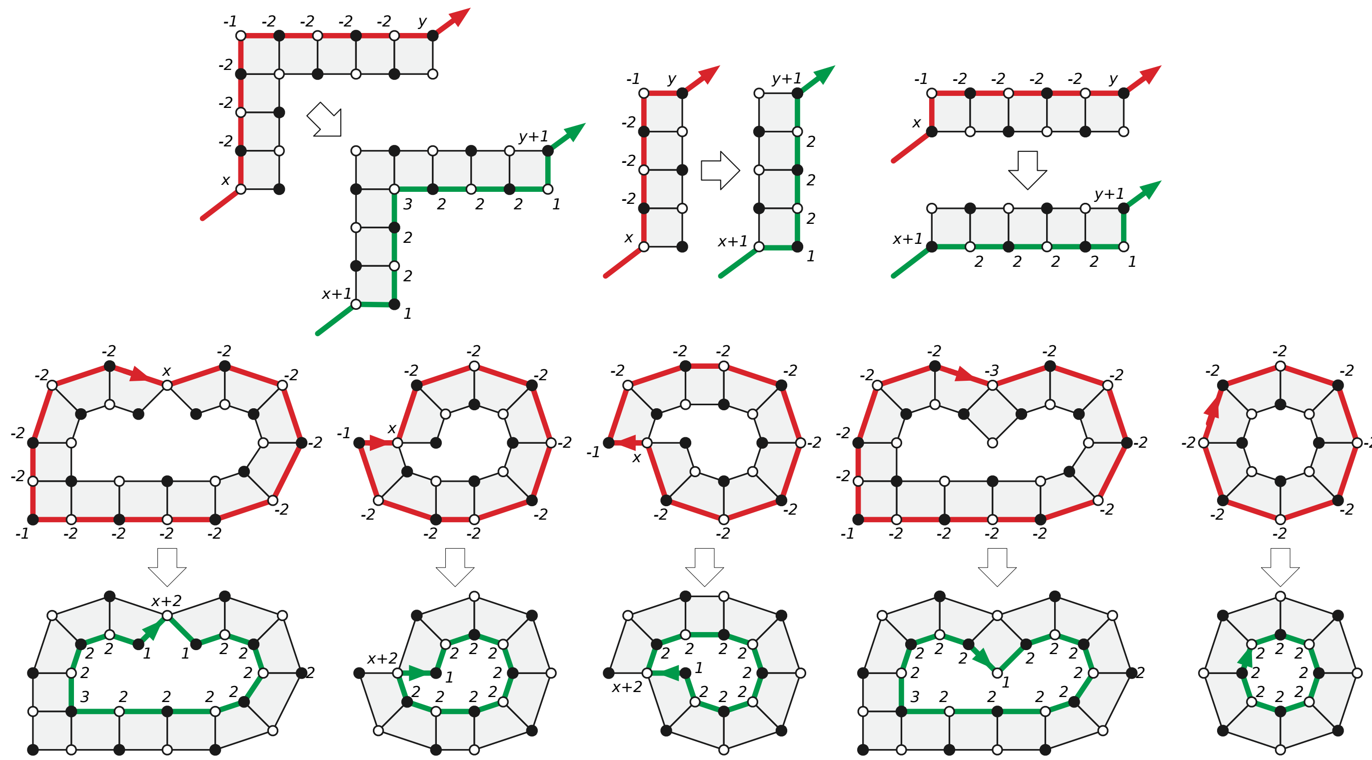

Figure 5. Elementary right shifts.

only to reduced cycles. We also have $x \neq \bar{2}$ and $y \neq \bar{2}$; otherwise, either the run-length encoding is incomplete, or we are considering the wrong case. Moreover, our assumption that $g \geq 2$ implies that $3 \neq \bar{1}$. Thus, applying an elementary right shift to a reduced cycle yields another reduced cycle with one less $\bar{1}$. It follows that a reduced cycle of length ℓ can be transformed into a canonical cycle in the same free homotopy class by a sequence of at most ℓ elementary right shifts; in fact, this sequence can be found and executed in $O(\ell)$ time.

Lemma 5.1. *Given a reduced cycle α of length ℓ in a system of quads, we can compute a canonical cycle freely homotopic to α in $O(\ell)$ time.*

Proof: As in the proof of Lemma 4.3, our algorithm computes and maintains the run-length-encoded turn sequence of α, together with the directed edges entering each run. The algorithm simply scans once through the compressed turn sequence. Whenever we encounter a $\bar{1}$, we eliminate it with an elementary right shift, which involves at most seven runs and thus requires only constant time. Since no elementary right shift introduces a new $\bar{1}$ into the turn sequence, no backtracking is required. □

5.2 Canonical Cycles are Unique

Finally, given two cycles α and β in a system of quads, each with length at most ℓ, we can decide whether α and β are freely homotopic as follows. First we compute canonical cycles α' and β' that are freely homotopic to α and β, respectively, using Lemmas 4.3 and 5.1. Then to determine whether α' and β' are equal as cycles, we break the cycles into loops by choosing arbitrary basepoints and then check whether the loop α' is a subpath of the loop $\beta' \cdot \beta'$ using any linear-time string-matching algorithm [4, 16, 25]. Finally, we report that α and β are freely homotopic if and only if α' and β' are the same cycle. The overall running time of our algorithm is clearly $O(\ell)$. It remains only to prove that our algorithm is correct.

Lemma 5.2. *There is exactly one canonical cycle in each free homotopy class in a system of quads.*

Proof: Let α and β be distinct *reduced* cycles in the same free homotopy class. It suffices to show that either α is not canonical or β is not canonical. Let A be a reduced annular diagram over Σ whose boundary cycles map to α and β, as guaranteed by Schupp's Lemma 2.2.

First, we claim that each boundary cycle of A is a *simple* cycle. Otherwise, there is a cut vertex x that separates A into a smaller annular diagram A' and a disk diagram Δ. Lemma 4.1 implies that the boundary of Δ has at least two brackets that do not touch A, contradicting our assumption that α and β are reduced. A similar argument implies that no interior edge of A has both endpoints on the same boundary cycle of A.

Next, we claim that A has no interior vertices. Suppose we assign an angle $1/4$ to every corner in A. Then every

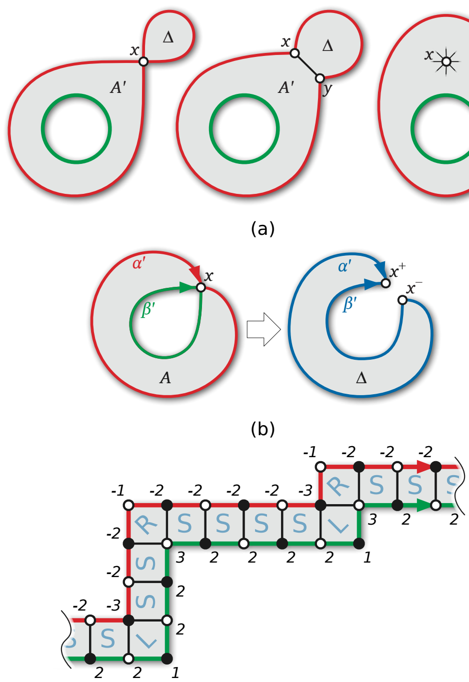

Figure 6. (a) Impossible features in an annular diagram with reduced boundary cycles. (b) Cutting an annular diagram into a disk diagram. (c) A cyclic staircase.

face of A has curvature zero, and every interior vertex of A has curvature at most $-1/2$. The Combinatorial Gauss-Bonnet Theorem implies that the sum of all vertex curvatures is zero. Thus, if A has at least one interior vertex, the total curvature of its boundary vertices is at least $1/2$, which implies that the boundary of A has at least two brackets, contradicting our assumption that α and β are reduced. See Figure 6(a).

Suppose the boundary cycles of A contain a common vertex x. Then we can consider the boundary cycles of A to be loops based at x; let α' and β' denote the images of these boundary loops under the diagram map $\delta : A \to \Sigma$. Thus, α' and β' are loops based at $\delta(x)$ that respectively traverse the cycles α and β exactly once.

Now consider the disk diagram Δ obtained by *cutting* the annular diagram A at x, as shown in Figure 6(b). The vertex x survives in Δ as two boundary vertices x^+ and x^-; the natural diagram map sends the boundary of Δ to the loop $\alpha' \cdot \bar{\beta}'$. Lemma 4.1 implies that the boundary of Δ has (without loss of generality) four left brackets. Because α and β are reduced cycles, each of these brackets must end at either x^+ or x^-. It follows that the loops α' and $\bar{\beta}'$ each have at least one interior vertex with turn 1, so β has a vertex with turn $\bar{1}$ and thus is not canonical. Similarly, if the boundary of Δ has four right brackets, then α is not canonical.

Finally, suppose the boundary cycles of A are disjoint. Then A must consist of a simple cycle of quadrilaterals, each with exactly two edges on the boundary of A. Without loss of generality, suppose the boundary of A is oriented so that α is the image of the left boundary cycle. Consider the inner dual cycle A^*, which has a node for each face of A and an edge for each interior edge of A, oriented consistently with the boundary cycles of A. Call each face of A a *left face*, a *straight face*, or a *right face*, depending on whether A^* turns left, moves straight ahead, or turns right inside that face; see Figure 6(c). Any two left faces must be separated by a right face and vice versa; otherwise, the boundary of A would contain a bracket, contradicting our assumption that α and β are reduced. In particular, A has the same number of left and right faces. If A has a right face, then the left boundary of A has a vertex with turn $\bar{1}$; otherwise, A has only straight faces, which implies that every vertex in α has turn $\bar{2}$. In either case, we conclude that α is not canonical. □

Theorem 5.3. *Given a combinatorial surface Σ with complexity n and two cycles α and β in Σ, each with length at most ℓ, we can determine whether α and β are freely homotopic in Σ in $O(n + \ell)$ time.*

References

[1] Thomas Banchoff. Critial points and curvature for embedded polyhedra. *J. Diff. Geom.* 1:245–256, 1967.

[2] Norman Biggs. Spanning trees of dual graphs. *J. Comb. Theory* 11:127–131, 1971.

[3] Ronald V. Book. Dehn's algorithm and the complexity of word problems. *Amer. Math. Monthly* 95(10):919–925, 1988.

[4] Robert S. Boyer and J. Strother Moore. A fast string searhcing algorithm. *Commun. ACM* 20(10):762–772, 1977.

[5] Martin R. Bridson and André Haefliger. *Metric Spaces of Non-Positive Curvature*. Grundlehren der mathematischen Wissenschaften 319. Springer-Verlag, 1999.

[6] Éric Colin de Verdière and Jeff Erickson. Tightening non-simple paths and cycles on surfaces. *SIAM J. Comput.* 39(8):3784–3813, 2010.

[7] Max Dehn. Über unendliche diskontinuierliche Gruppen. *Math. Ann.* 71(1):16–144, 1911.

[8] Max Dehn. Transformation der Kurven auf zweiseitigen Flächen. *Math. Ann.* 72(3):413–421, 1912.

[9] Tamal K. Dey and Sumanta Guha. Transforming curves on surfaces. *J. Comput. System Sci.* 58:297–325, 1999.

[10] Tamal K. Dey and Haijo Schipper. A new technique to compute polygonal schema for 2-manifolds with application to null-homotopy detection. *Discrete Comput. Geom.* 14(1):93–110, 1995.

[11] Bernard Domanski and Michael Anshel. The complexity of Dehn's algorithm for word problems in groups. *J. Algorithms* 6(4):545–549, 1985.

[12] Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. American Mathematical Society, 2010.

[13] David Eppstein. Dynamic generators of topologically embedded graphs. *Proc. 14th Ann. ACM-SIAM Symp. Discrete Algorithms*, 599–608, 2003. ArXiv:cs.DS/0207082.

[14] David B. A. Epstein, James W. Cannon, Derek F. Holt, Silvio V. F. Levy, Michael S. Paterson, and William P. Thurston. *Word Processing in Groups*. Jones & Bartlett, 1992.

[15] David B. A. Epstein and Derek F. Holt. The linearity of the conjugacy problem in word-hyperbolic groups. *Int. J. Alg. Comput.* 16:287–305, 2006.

[16] Zvi Galil. On improving the worst case running time of the Boyer-Moore string matching algorithm. *Commun. ACM* 22(9):505–508, 1979.

[17] Steve M. Gersten and Hamish B. Short. Small cancellation theory and automatic groups. *Invent. Math.* 102:305–334, 1990.

[18] Étienne Ghys and Pierre de la Harpe, editors. *Sur les groupes hyperboliques d'après Mikhael Gromov*. Progress in Mathematics 88. Birkhäuser, 1990.

[19] Mikhail Gromov. Hyperbolic groups. *Essays in Group Theory*, 75–265, 1987. Math. Sci. Res. Inst. Pub. 8, Springer-Verlag.

[20] Yusuke Higuchi. Combinatorial curvature for planar graphs. *J. Graph Theory* 38(4):220–229, 2001.

[21] Derek F. Holt. Word-hyperbolic groups have real-time word problem. *Int. J. Alg. Comput.* 10:221–227, 2000.

[22] Minoru Ishida. Pseudo-curvature of a graph. Lecture, Workshop on Topological Graph Theory, Yokohama National Univ., 1990. Cited in [20].

[23] Egbert R. van Kampen. On some lemmas in the theory of groups. *Amer. J. Math.* 55(1/4):268–273, 1933.

[24] Felix Klein. Letter to Henri Poincaré, May 14 1882. *Gesammelte Mathematische Abhandlungen* 3, 615–616. Cited in [38].

[25] Donald Knuth, James H. Morris, Jr., and Vaughan Pratt. Fast pattern matching in strings. *SIAM J. Comput.* 6(2):323–350, 1977.

[26] Francis Lazarus and Julien Rivaud. On the homotopy test on surfaces. *Proc. 53rd Ann. IEEE Symp. Foundations Comput. Sci.*, to appear, 2012. ArXiv:1110.4573.

[27] Roger C. Lyndon. On Dehn's algorithm. *Math. Ann.* 166:208–228, 1966.

[28] Roger C. Lyndon and Paul E. Schupp. *Combinatorial Group Theory*. Classics in Mathematics. Springer-Verlag, 2001.

[29] Jon McCammond and Daniel Wise. Fans and ladders in combinatorial group theory. *Proc. London Math. Soc.* 84:499–644, 2002.

[30] Bojan Mohar and Carsten Thomassen. *Graphs on Surfaces*. Johns Hopkins Univ. Press, 2001.

[31] Alexander Yu. Ol'shanskii. *Geometry of Defining Relations in Groups*. Nauka Publ., 1989. In Russian. English translation in [32].

[32] Alexander Yu. Ol'shanskii. *Geometry of Defining Relations in Groups*. Kluwer Acad. Publ., 1991. Translated by Yurii A. Bakhturin.

[33] Henri Poincaré. Cinquième complement à l'analysis situs. *Rendiconti del Circulo Matematico di Palermo* 18:45—110, 1904. English translation in [34].

[34] Henri Poincaré. *Papers on Topology:* Analysis Situs *and Its Five Supplements*. History of Mathematics 37. American Mathematical Society, 2010. Translated from the French and with an introduction by John Stillwell.

[35] George Pólya. An elementary analogue to the Gauss-Bonnet theorem. *Amer. Math. Monthly* 61:601–603, 1954.

[36] Haijo Schipper. Determining contractibility of curves. *Proc. 8th Ann, Symp. Comput. Geom.*, 358–367, 1992.

[37] Paul E. Schupp. On Dehn's algorithm and the conjugacy problem. *Math. Ann.* 178:119–130, 1968.

[38] John Stillwell. The word problem and the isomorphism problem for groups. *Bull. Amer. Math. Soc.* 6(1):1982, 1982.

[39] John Stillwell. *Classical Topology and Combinatorial Group Theory*, 2nd edition. Graduate Texts in Mathematics 72. Springer-Verlag, 1993.