ISSN 1364-0380 **175**

Geometry & Topology
Volume 2 (1998) 175–220
Published: 4 January 1999

A new algorithm for recognizing the unknot

Joan S Birman
Michael D Hirsch

Math Dept, Columbia University, NY, NY 10027, USA
Math and CS, Emory University, Atlanta, GA 30322, USA

Email: `jb@math.columbia.edu` and `hirsch@mathcs.emory.edu`

Abstract

The topological underpinnings are presented for a new algorithm which answers the question: "Is a given knot the unknot?" The algorithm uses the braid foliation technology of Bennequin and of Birman and Menasco. The approach is to consider the knot as a closed braid, and to use the fact that a knot is unknotted if and only if it is the boundary of a disc with a combinatorial foliation. The main problems which are solved in this paper are: how to systematically enumerate combinatorial braid foliations of a disc; how to verify whether a combinatorial foliation can be realized by an embedded disc; how to find a word in the the braid group whose conjugacy class represents the boundary of the embedded disc; how to check whether the given knot is isotopic to one of the enumerated examples; and finally, how to know when we can stop checking and be sure that our example is *not* the unknot.

AMS Classification numbers Primary: 57M25, 57M50, 68Q15

Secondary: 57M15, 68U05

Keywords: Knot, unknot, braid, foliation, algorithm

Proposed: David Gabai Received: 3 July 1997
Seconded: Wolfgang Metzler, Cameron Gordon Revised: 9 January 1998

1 Introduction

The goal of this manuscript is the development of a new algorithm to answer the question: Given a knot K which is defined by a diagram, does K represent the unknot? Our algorithm is suitable for computer enumeration. Its approach is straightforward:

(1) We show how to construct a sufficiently large set of diagrams which represent the unknot;

(2) We introduce a complexity function which allows us to order these diagrams, as we construct them, in order of complexity;

(3) We learn how to test in a systematic way whether an arbitrary diagram for a knot K is equivalent to one of the diagrams on the list;

(4) We arrange that the checking process stop in a finite time.

The focus of this paper will be on the topological underpinnings of the algorithm. We are in the process of implementing the algorithm, and of assembling computer-generated data for the ordered list which we produce, and the data should be interesting. We plan to write a second paper on that work, when it is complete and done efficiently enough to give us the data we would like to see.

Before we describe our approach, we give a brief review of related work on the problem.

- In the 1960's W Haken [8] used the concept of a normal surface to show that the homeomorphism problem is solvable for triangulated 3–manifolds which contain an incompressible surface. The unknot K bounds a disc, and the disc is an incompressible surface in the 3–manifold which is obtained by removing an open tubular neighborhood of K from S^3, so Haken's work proves the existence of an algorithm for recognizing the unknot. See [9] for a review of Haken's contribution to the problem.
- Several months after an earlier draft of this paper was submitted for publication Hass, Lagarias and Pippenger announced new results on the problem of recognizing the unknot in [10]. Their work investigates Haken's algorithm in detail, using more recent contributions of Jaco and Oertel and new techniques from computer science to sharpen Haken's algorithm and place explicit bounds on its running time. They then go on to prove that the problem of recognizing the unknot is in class NP. That issue is somewhat tangential to the subject matter of this paper. The emphasis of this paper is on the existence of a new algorithm for unknottedness,

rather than the complexity. Our proof that we can stop checking after an explicit finite time will use similar methods to [10], but the algorithm itself is totally different. It is not clear at this writing whether our algorithm demonstrates that the problem of detecting unknots is in NP.

- In a different direction, an open conjecture is whether there exists a non-trivial knot whose Jones polynomial is 1. If none exists then the Jones polynomial detects the unknot.

Our work is in the setting of closed braids. When we say that we list 'a sufficiently large set of diagrams which represent the unknot' we mean that we enumerate the conjugacy classes of the unknot in the braid group B_n, where n is the number of Seifert circles in the given diagram of K, in an appropriate order. Since there is a very simple algorithm [13] to change every knot diagram with n Seifert circles to a closed n–braid diagram and to read off a representing open braid, and since the conjugacy problem in the braid group is a solved problem ([7],[6],[4]) we will then have the tool we need to solve problem 3, ie to test (one conjugacy class at a time) whether the closed n–braid which we constructed from the given diagram of K is conjugate to one of the members of our list. The list is infinite for $n \geq 4$, and our main problems are to construct the list, to order it in an appropriate way, and to learn when we can stop testing. Those are non-trivial problems, and we will bring much structure to bear on them, in addition to using the known solution to the conjugacy problem.

The unknot is the unique knot which bounds a disc, and our tool for enumerating its closed n–braid representatives is based on the combinatorics of certain *braid foliations* of the disc. These foliations were introduced by D Bennequin in [1]. They were studied systematically as a tool in knot theory by the first author and W Menasco in a series of papers with the common title 'Studying links via closed braids', for example see [5]. Our detailed work involves many ideas from those papers, but for convenience our references will be mainly to the review article [3], which gathers together in one place the machinery developed in those papers. Our technique for enumerating all closed braid representatives of the unknot is in fact implicit in D. Bennequin's work [1]. It is a method of "stabilizing" a complicated embedded disc to obtain a simpler one whose boundary has much higher braid index. We use the reverse of this procedure to generate complicated discs whose boundaries have low braid index. Some of these are not embeddable, so we develop a new (and surprisingly simple) test for embeddability, allowing us to eliminate any non-embeddable ones that may have arise in the course of the enumeration.

Having in hand an embeddable foliated disc with associated combinatorial data, we know (from a theorem proved in [3]) that its embedding in 3–space relative to cylindrical coordinates is determined up to foliation-preserving isotopy. However, we still have to solve the problem of determining a word in the generators of the braid group which represents the boundary of the given disc. The solution to that problem is new to this paper and turns out to be quite elegant. All of this, combined with the solution to the conjugacy problem in B_n in [4], solves problems 1, 2 and 3 above.

The complexity measure which we assign to our foliated disc is a pair of integers (n, v), where n is the braid index of the boundary and v is the number of times the braid axis intersects the disc. The given knot K is defined by a diagram, and as noted earlier n is also the number of Seifert circles in the diagram, which is thus fixed for each example. To find a bound on v we must relate v to the crossing number k of K. For this part of the argument we construct a triangulation of the complement of a tubular neighborhood of K, doing it so that the braid axis meets the interiors of exactly 4 tetrahedra, in a controlled way. We then find an upper bound on the number t of tetrahedra, as a function of k and n. As is well-known, the work of Kneser and Haken implies the existence of an upper bound on the number of times the disc we are seeking can intersect a single tetrahedron. Fortunately we do not need to compute that bound because Lemma 6.1 of [10] does the job for us. Multiplying by 4 we obtain an upper bound for v, which then tells us when we can stop testing.

Here is an outline of the paper. In Section 2 we review the prerequisite material about braid foliations. This section is without proofs, but the material is fairly understandable and believable. A convenient reference, complete with proofs, is available [3]. In Section 3 we show how to test whether a given combinatorially foliated surface actually corresponds to an embedded foliated surface, and if so how to find a braid word which represents the boundary. In Section 4 we show how to enumerate the ordered list of closed n–braid representatives of the unknot which is the basis for our algorithm. In Section 5 we prove our 'halting theorem'. In Section 6 we present the algorithm.

For completeness, we show in Appendix A how to rapidly modify an arbitrary knot diagram to a closed braid diagram, with control over the extra crossings which are added. This part of the algorithm is based upon the work of Vogel, reported on in [13]. In Appendix B we review the solution to the conjugacy problem in B_n which we are using in our algorithm. The theoretical basis for that algorithm is established in [4].

Acknowledgements We thank Elizabeth Finkelstein for her many contributions to this paper, both through her work in [3] and through our discussions with her at an early stage in this work. We thank Joel Hass for helpful conversations. It was only after we read the manuscript [10] and talked to him that we understood how to give the proof of the Halting Theorem which is presented here, replacing a much more awkward solution to that problem in the earlier draft of this paper. We also thank William Menasco and Brian Mangum for helpful conversations.

The first author acknowledges partial support from the following sources: the US National Science Foundation, grants DMS 94-02988 and DMS 97-05019; Barnard College, for salary support during a Senior Faculty Research Leave; the Mathematical Sciences Research Institute, where she was a Visiting Member when part of this work was done; and the US Israel Binational Science Foundation. The second author would like to thank the US National Science Foundations for partial support under grants ASC-9527186 and DMS-9404261.

2 Braid foliations of spanning surfaces for knots

In this section we will review the basic theorems about the braid foliations associated to an incompressible orientable surface spanning a knot or link. The main theorems in this field are collected in the survey paper [3], which we will use as our preferred reference for the material of this section.

While we are interested mainly in closed braid representatives of the unknot, we state everything in terms of links, and the (possibly disconnected) incompressible spanning surfaces of genus $g \geq 0$ which they bound. Admitting these complications requires almost no extra work and yields much simplified proofs in Section 3. It is also necessary for any future extensions of these ideas to true (knotted) knots.

Choose cylindrical coordinates (r, θ, z) in 3–space (S^3 thought of as $\mathbb{R}^3$ union a point at ∞). A link K is a *closed braid* with the z–axis as its *braid axis* if each component of K has a parameterization $\{(r(t), \theta(t), z(t)) : t \in [0,1]\}$ such that $r(t) \neq 0$ and $d\theta/dt > 0$ for all $t \in [0,1]$. This means that K intersects each half-plane H_θ through the axis transversally, as in Figure 1. It follows that the number of points in $K \cap H_\theta$ is independent of θ. This number is the *braid index*, $n = n(K)$, of K. Notice that the closed 4–braid diagram in Figure 1 contains 4 Seifert circles. See the Appendix for a proof that any diagram with n Seifert circles can be modified to a closed n–braid, adding a controlled number of crossings.

Let K be a closed braid and let F be an embedded orientable surface of minimal genus spanned by K. Note that both F and ∂F may be disconnected. Let

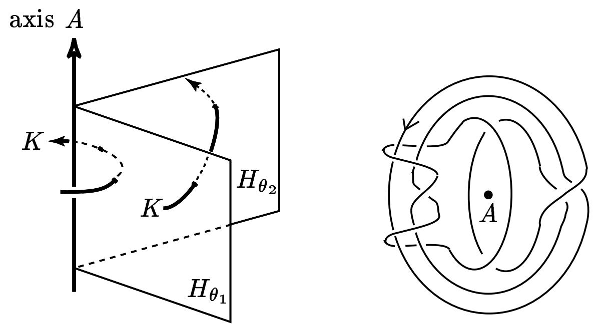


Figure 1: Braids and closed braids

$\mathcal{H} = \{H_\theta : \theta \in [0, 2\pi]\}$ be the open book decomposition of $\mathbb{R}^3$ by half-planes with boundary on the z–axis. Assume that F is in general position with respect to $\mathcal{H}$ and consider the induced foliation on F. This foliation is the *braid foliation* of F.

The braid foliation is singular both at points where F meets the braid axis A, and at points where F is tangent to leaves of $\mathcal{H}$. By general position, these latter type of singularities can be assumed, *a priori*, to be of saddle type, or center type with neighborhoods foliated by circles. The following is a restatement of results of Bennequin [1]. It classifies the leaves and singularities of the braid foliation after an isotopy.

Theorem 2.1 ([3], Theorems 1.1,1.2) *After a modification of F rel ∂F (by isotopy when ∂F is non-split) the following hold:*

(1) *The braid foliation near ∂F (see Figure 2) is transverse to ∂F. It is radial in a neighborhood of each point of $A \cap F$ (see Figure 3).*

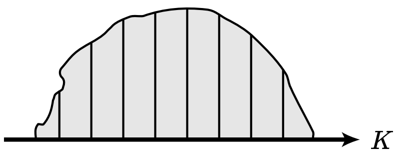


Figure 2: The braid foliation in a neighborhood of ∂F.

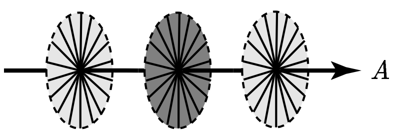


Figure 3: The braid foliation in a neighborhood on F of each point of $A \cap F$.

(2) *The non-singular leaves of the braid foliation fall into three types: a, b and c. (See the left sketch in Figure 4.)*

a: *arcs with one endpoint on K and one on A, and*

b: *arcs with both endpoints on A.*

c: *arcs with both endpoints on K.*

However arcs of type c are necessarily singular because intersections of K with every fiber of $\mathcal{H}$ are coherently oriented (see the right sketch in Figure 4), so they do not occur as non-singular leaves.

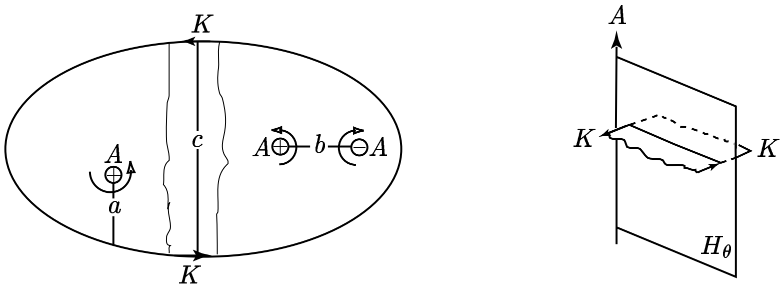


Figure 4: Leaves of type a,b and c

(3) *The singular points of the braid foliation are of two types which we call vertices and singularities:*

Vertices: points of intersection between F and A. The foliation is radial near the vertices (see Figure 3).

Singularities: A point of tangency between F and some H_θ. These tangencies are simple saddles (see Figure 5). Such H_θ are called singular. The point of tangency is called a singularity. The singularity, together with its four leaves, is called a singular leaf. The four leaves of a singular leaf are called branches of the singular leaf.

(4) *Singularities fall into three types: aa, ab and bb (see Figure 5).*

aa: *those singularities between two a–arcs,*

ab: *those between an a–arc and a b–arc, and*

bb: *those between two b–arcs.*

(5) *The vertices are (circularly) ordered by their order on the braid axis A. After isotopy distinct singular leave are on distinct H_θ and are thus circularly ordered, as well.*

In part 2 of the theorem, a salient point is that non-singular circular leaves can occur at local minima and maxima, and the theorem says that (subject to the assumption that F has minimum genus among all orientable surfaces bounded

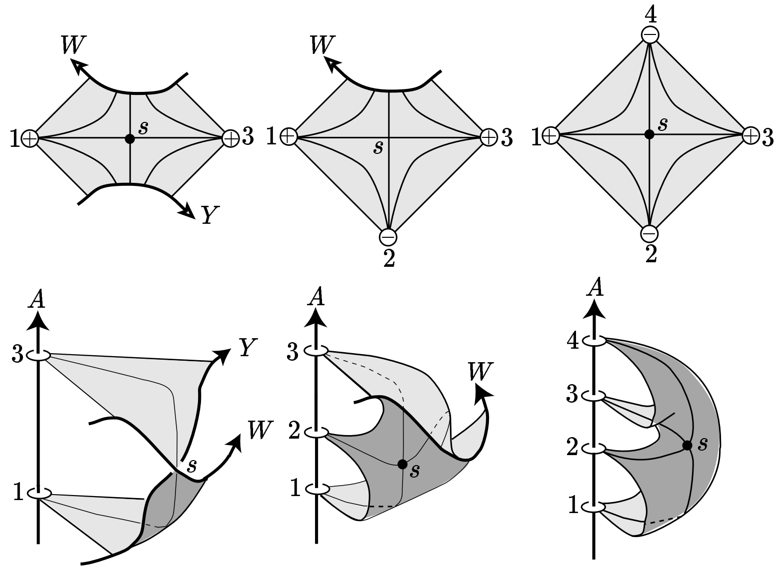


Figure 5: The three singularity types

by K) we can cut off any maxima and fill up the minima without leaving the class of surfaces which are of interest. If there no local minima or maxima, only the vertex and saddle type singularities of part 3 can occur. This arrangement, together with the fact that a leaf with both of its endpoints on K is necessarily singular, is responsible for the rich combinatorics of braid foliations.

There are additional combinatorial data. Since the original braid K was oriented, F has an orientation. If that orientation agrees with the orientation of A at a vertex v, (ie the normal vector has positive inner product with the oriented tangent vector to A) then we say the vertex is *positive*, otherwise it is *negative*. Similarly, at a singularity, if the normal vector is positive with respect to $d\theta$ we say the singularity is *positive*, otherwise it is *negative*.

There is dual viewpoint with regard to these foliations. Instead of focusing on singular leaves, one can break the surface up into *tiles*, one for each singularity, by cutting along appropriate a–arcs and b–arcs. Figure 5 shows the tile types, for each singularity. This is the point of view in [3], and so we use the terminology “tiled surface”.

We will need the following facts about singularities. They are self evident from Figure 5 and follow easily from orientation considerations and the previous theorem. Each singularity is connected to exactly two positive vertices along non-adjacent branches of the singular leaves. Each of the other two leaves can go to either the boundary of the orientable surface, or to a negative vertex.

Definition A *tiled surface* $\mathcal{F}$ is a 3–tuple (F, G, C) where F is an oriented surface, G is a graph which is embedded in F (so that it has a well-defined neighborhood in F), with some additional combinatorial data C which we call *decorations*. The graph should be of the type attainable as the graph of singular leaves, ie, G must be tripartite with each node either a *vertex*, a *singularity*, or a *boundary point*. The singularities are of index 4 and are adjacent to at most two boundary points (on non-adjacent edges). The boundary points are on the boundary of F and adjacent to a single singularity. Each component $F' \subset F \setminus G$ is a disc, also $\partial F'$ contains either exactly 4 edges of G or exactly 3 edges, one of which meets F' on both sides. Each vertex has a sign, as does each singularity. The vertices are circularly ordered, as are the singularities. Vertices of the same sign can be adjacent to the same singularity only if they are on edges which are non-adjacent at the singularity.

We wish to emphasize the decorations of the tiled surface. In the tiled surface literature these decorations have been largely ignored, or, at best, implicit. Tiled surfaces were studied primarily in terms of the graph with only secondary thought given to the decorations.

If $\mathcal{F}$ is a tiled surface, then F can be foliated (uniquely, up to homeomorphism) by a singular foliation so that G is the union of singular leaves. Thus every tiled surface as defined above is implicitly foliated by a–arcs and b–arcs as specified in Theorem 2.1. Traditionally, one uses the foliation instead of the graph. Graphs are more natural to use in an algorithmic context, and make the decorations easier to understand, so we use them in our definition.

It is natural to abuse notation and think of $\mathcal{F}$ as a surface, rather than as a tuple (F, G, C); we will try not to do this. We shall consistently use the convention that the symbol for the surface will be a roman capital and the tiled surface will be the same letter in a calligraphic font.

Definition An *embeddable tiled surface* is a tiled surface which is actually achieved as the graph of singular leaves of some embedded orientable surface with closed braid boundary. This embedding is essentially unique. See Theorem 2.2.

One more concept from the previously cited papers of Bennequin, Birman and Menasco will be helpful, because it gives a very simple way to exclude unwanted examples. As before, we refer the reader to [3] for a detailed exposition. Consult Figure 6.

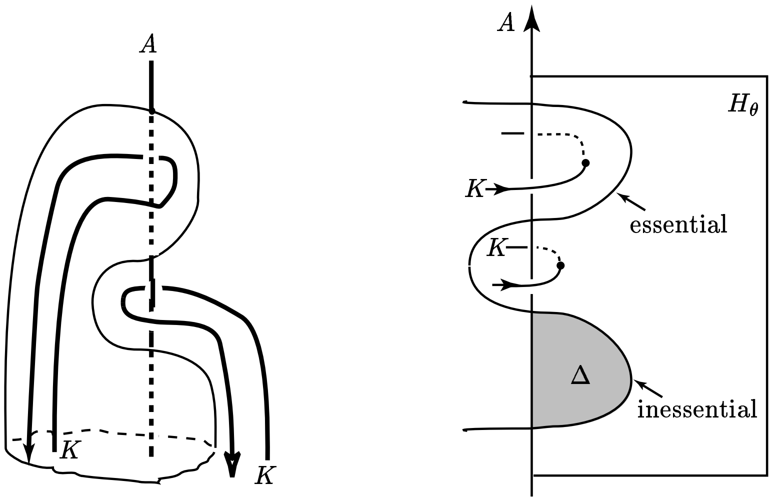


Figure 6: Essential and inessential b–arcs

The sketch on the left illustrates a 'pocket' in an embedded disc. It cannot be removed because the knot is in the way. If the knot was not an obstruction, we could eliminate the pocket (and remove two vertices in the tiling) by an isotopy. This leads us to a definition.

Definition A b–arc β is said to be *essential* if both sides of H_θ split along β are pierced by K. See the right sketch in Figure 6. An embeddable tiled surface is an *essential tiled surface* if all the b–arcs of the braid foliation induced by the embedding of the tiled surface are essential. An embeddable tiled surface which is not an essential tiled surface is said to be an *inessential tiled surface.*

Example Figure 7 gives an example of a tiled surface of genus 2. We will see shortly how to test that it is embeddable and essential. The 11 singularities are indicated by small black dots, signed and labelled $a, b, c, d, e, f, g, h, i, j, k$ to correspond to their cyclic order in the fibers around the axis. The 8 white circles (6 positive and 2 negative) are the signed tile vertices. They are labelled $1, 2, 2.1, 2.2, 3, 4, 5, 6$ to them describe their order on the braid axis. (It will become clear as we proceed why we choose non-integer labels for the negative vertices).

In the next section we will learn how to test whether a given example is embeddable. We are aided in that project by the fact that when a tiled surface is embeddable, then there is a unique embedding, up to foliation preserving isotopy:

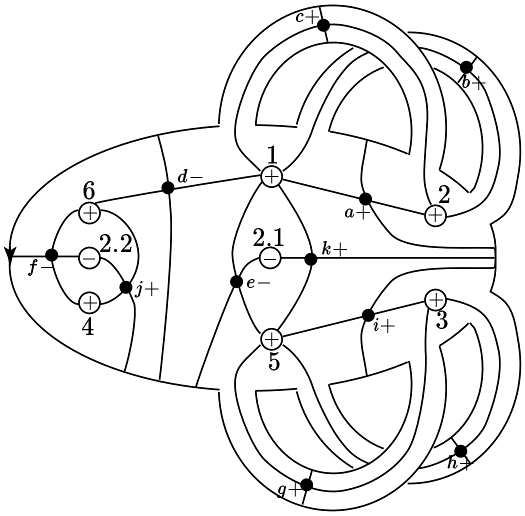


Figure 7: An example of an embeddable essential tiled surface

Theorem 2.2 [3, Theorem 4.1] *The combinatorial data for a embeddable tiled surface $\mathcal{F}$, ie, the embedded graph G and its embedding in F, the circular ordering for its vertices, the circular ordering for its singularities, and the signs of the vertices and singularities, determine the embedding in S^3. This embedding is unique up to foliation preserving isotopy. The embedding of the boundary is determined by the same data, restricted to singular leaves which meet the boundary and their associated vertices.*

3 Testing for embeddability and finding the boundary word

Given a knot or link, it is natural to ask what surfaces it spans. In this section we study a dual question: Given a tiled surface, is it embeddable? And if it is embeddable, what braid is represented by its boundary? Our main results on these matters are Theorems 3.4 and 3.5. During most of the section we will ignore the question of whether the surface is essential, but at the end of the section Proposition 3.6 will give a very simple test which can be used to eliminate inessential tiled surfaces.

3.1 A special case: positive tiled surfaces

Our work begins with a special case of the embeddability question: when is a tiled surface which has only positive vertices embeddable in 3–space? The answer, roughly, is "most of the time":

For convenience, we call a tiled surface with only positive vertices a *positive* tiled surface. For such a surface every singularity is type aa and each singularity is connected to exactly two vertices along non-adjacent branches of the singular leaves (see Figure 5). Figure 8(a) is an example of a positive tiled surface. The example is very simple, and so it's easy to understand the embedding in 3–space which is given in Figure 8(b). The surface is depicted as a Seifert surface for the closed braid σ_1^3, in Artin's well-known generators of the braid group. The two discs have been arranged as concentric discs in 3–space, with disc 2 above disc 1. The two discs are joined by three half-twisted bands. The singularities in all 3 bands are negative. The boundary is the negative trefoil knot.

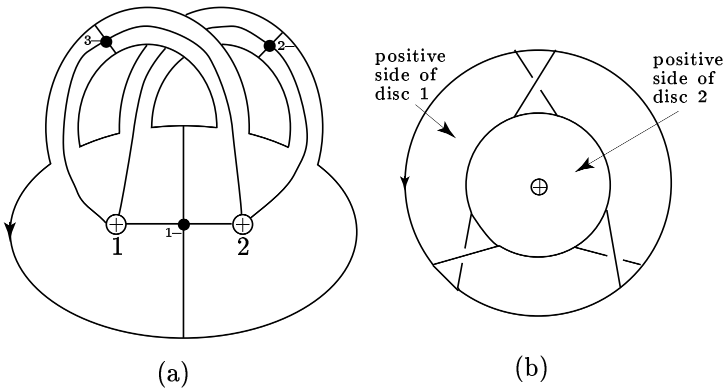


Figure 8: Example of a positive tiled surface

Lemma 3.1 *Let $\mathcal{F} = (F, G, C)$ be a positive tiled surface. Assume that the combinatorial data C is subject to a single restriction: the cyclic order of the singularities around each vertex of valence ≥ 3 is counterclockwise when viewed on the positive side of $\mathcal{F}$. Then $\mathcal{F}$ is embeddable.*

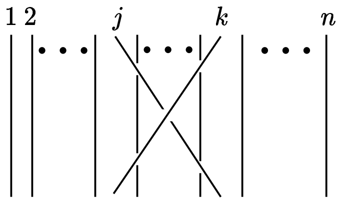
Proof Clearly, in an embedded surface the singular leaves meeting at a vertex are circularly ordered because the ordering is given precisely by their order around the braid axis. Thus the order condition of the lemma is necessary. We need to show it suffices when there are no negative vertices in G. Assuming the order condition is met about each vertex, we shall construct an embedded orientable surface in three space whose boundary is a braid, and whose graph of singular leaves in the associated braid foliation is isomorphic to G with an isomorphic embedding and isomorphic combinatorial data.

Let $v_1, v_2, \ldots, v_P$ be the vertices of $\mathcal{F}$, and let $s_1, s_2, \ldots s_S$ be the singularities, written in order. Since $\mathcal{F}$ has no negative vertices, each singularity is adjacent to exactly two vertices with the other two singular leaves going to $\partial\mathcal{F}$. Let δ_i be a disc parallel to the xy-plane, centered on the z-axis, height i, and radius $1/i$. If s_i is adjacent to v_j and v_k, then connect the discs δ_j and δ_k with small twisted bands at angle $2\pi i/S$. The twisted band can twist in either of two ways. Choose the twist so that the part connected to the positive part of δ_i is positively oriented with respect to $d\theta$ if the sign s_i is positive, and choose the other twist if s_i is negative. Note that the edges of the twisted band can be made arbitrarily close to straight lines because $1/i$ is a convex function.

Clearly, then, the surface F^e given by the δ_i and the twisted bands is a surface with closed braid boundary and associated graph G. The surface F^e is orientable since a twisted band always connects the discs so that the upper sides connect to each other. All that remains is to check the signs of the vertices and singularities. Clearly all the vertices are positive and the twists were chosen so that the signs of the singularities would agree. Thus the braid foliation on F^e and $\mathcal{F}$ have the same combinatorial data, and it then follows from Theorem 2.2 that $\mathcal{F}$ is embeddable. □

We next consider the question of determining a braid word which describes the boundary of a positive embeddable tiled surface, ie an embeddable tiled surface which has only positive vertices in its foliation. Since isotopic embeddings of the same tiled surface can have boundaries that differ by a conjugation, the answer can only be determined up to conjugation. There is a convenient set of generators for the braid group known as *band generators*. They are particularly useful in algorithmic questions, having been to give fast solutions to the word problem in B_n [4] and the conjugacy problem in B_3 in [14] and B_4 in [12].

Let k, j be integers with $n \geq k \geq j \geq 1$. Let $a_{k,j}$ denote an elementary braid in which strands k and j exchange places with strand k crossing over strand j and with both passing in front of all intermediate braid strands. See Figure 9. The collection of elementary braids $a_{k,j}$ and their inverses clearly generate B_n because they contain as a subset the Artin generators $\sigma_i = a_{i+1,i}$. These are the *band generators* of the braid group. (See Appendix B for a discussion of these generators and their relations.)



The braid $a_{k,j}$

Figure 9: Band generators for the braid group

Lemma 3.2 *Let $\mathcal{F}$ be a positive embeddable tiled surface with P positive vertices (so its boundary is a braid with P strands) and S singularities at angles $\theta_1, \theta_2, \ldots, \theta_S$ of signs $\epsilon_1, \epsilon_2, \ldots, \epsilon_S$. Notice that every singularity in the embeddable tiled surface has a exactly two branches connected to positive vertices. For the singularity which is at angle θ_i, let v_{j_i}, v_{k_i} be the vertices associated to this singularity, where $k_i > j_i$. Then the closed braid given by $B(\mathcal{F}) = \prod_{i=1}^{S-1} a_{k_i,j_i}^{\epsilon_i}$ is a representative of $\partial\mathcal{F}$.*

Proof The proof follows directly from the construction which was given in the proof of Lemma 3.1. That lemma constructed an embedded surface with the same tiling as $\mathcal{F}$. We abuse notation slightly and call the embedded surface $\mathcal{F}$ as well.

From the construction we know that $\mathcal{F}$ is made of discs δ_i centered on the braid axis with twisted bands between these discs. At θ_i there is a band between δ_{j_i} and δ_{k_i} with a twist corresponding to the sign ϵ_i of the singularity at θ_i. The boundary in a neighborhood of θ_i is then exactly given by the band generator $a_{k_i,j_i}^{\epsilon_i}$. Thus the full closed braid is $B(\mathcal{F}) = \prod_{i=1}^{S-1} a_{k_i,j_i}^{\epsilon_i}$. □

3.2 The general case: finding the boundary word

We proceed to the general case, where both positive and negative vertices occur. The most efficient way to proceed is to bypass (for the moment) the question of how to test for embeddability, and assume that we have been given an embeddable tiled surface.

Definition Let $\mathcal{F}$ be an embeddable tiled surface with P positive vertices $v_1, v_2, \ldots, v_P$, in that order on A, and N negative vertices and S singularities at angles $\theta_1, \theta_2, \ldots, \theta_S$ of signs $\epsilon_1, \epsilon_2, \ldots, \epsilon_S$. Let μ be the number of components in $\partial\mathcal{F}$. Recall that every singularity in the foliation has exactly two branches connected to positive vertices. For the singularity which is at angle θ_i, let v_{j_i}, v_{k_i} be the orientable surfaces which are associated to these two vertices, where $k_i > j_i$. Let $EB(\mathcal{F})$ be the closed braid given by

$$EB(\mathcal{F}) = \prod_{i=1}^{S-1} a_{k_i,j_i}^{\epsilon_i}.$$

This is a link of one or more components, in 3–space. The word $EB(\mathcal{F})$ is called the *extended boundary word* of $\mathcal{F}$. If the tiled surface $\mathcal{F}$ has only positive vertices, then by Lemma 3.2 $EB(\mathcal{F}) = B(\mathcal{F})$. The word $EB(\mathcal{F})$ is given as a word in the band generators of the braid group B_P, where P is the number of positive vertices in the tiling. Our first lemma tells us that ∂F is represented by a word in the braid group B_{P-N}.

Lemma 3.3 *Let $\mathcal{F}$ be an embeddable tiled surface which has P positive and N negative vertices. Then the braid index of ∂F is $n = P - N$.*

Proof The braid index n is the linking number of $K = \partial F$ with the braid axis A. Linking number may also be computed as the algebraic intersection number of A with a surface which K bounds, ie $P - N$. □

The next theorem tells us that one of the components of the closed braid $EB(\mathcal{F})$ represents the boundary of the surface F.

Theorem 3.4 *Let $\mathcal{F} = (F, G, C)$ be an embeddable tiled surface with connected boundary, P positive vertices and N negative vertices. Let K' be the link represented by the extended boundary word $EB(\mathcal{F})$. Then K' is a link with $N+1$ components, at least N of which are closed 1–braids which are geometrically unlinked from the other components of K'. Let K be the link which is obtained from K' after deleting N 1–braid components of K'. Then K is a $(P-N)$–braid whose closed braid has 1 component, and this component represents ∂F.*

Proof Let $\mathcal{F}' = (F', G', C')$ be the tiled surface induced by removing a small neighborhood about each negative vertex in F and deleting the corresponding vertices from G and C. The surface F' is F *with N holes.* The tiled surface $\mathcal{F}'$ is clearly embeddable for the following reason: It a subset of the embeddable tiled surface $\mathcal{F}$. Also $\mathcal{F}'$ has no negative vertices, so Lemma 3.2 applies, and $\partial F'$ is represented by $B(\mathcal{F}')$. Thus $K' = \partial \mathcal{F}'$ is described by the word $B(\mathcal{F}')$. Notice that $B(\mathcal{F}')$ is identical with $EB(\mathcal{F})$, but not with $B(\mathcal{F})$.

Thinking of F' as a subset of F embedded in 3–space, we see that the boundary link of F' contains N small circles which bound discs each containing a single negative vertex in F. These discs are disjoint from F' (except on the boundary, of course), thus these N components of $\partial F'$ are geometrically unlinked from the other components. Except for these N components, the boundaries of F and F' are identical. Deleting these N components from $\partial F'$ yields exactly ∂F. □

Example We illustrate Theorem 3.4, using the example in Figure 7. The singularities are at $a, b, c, d, e, f, g, i, j, k$. Of those, only the singular leaves at a, b, c, g, h, i have two endpoints on ∂F. Assuming that our tiled surface is embeddable, we determine its extended boundary word. The tiling has 6 positive vertices, 2 negative vertices and 11 singularities. The extended boundary word $EB(\mathcal{F})$ is a 6–braid of length 11 in the band generators. It is:

$$EB(\mathcal{F}) = a_{2,1}^3 a_{6,1}^{-1} a_{5,1}^{-1} a_{6,4}^{-1} a_{5,3}^3 a_{6,4} a_{5,1}.$$

Figure 10 shows that it has 3 components, two of which (sketched as dotted curves) are closed 1–braids which are unlinked from the rest of the braid and from one-another. (It is *not* unique to this example that one has to check carefully to be sure that they are unlinked from the rest of the braid and from one-another.) The third component represents the boundary of the surface of Figure 7. Its braid index, which is the linking number of the axis A with ∂F, may be computed as the number of times the axis pierces $\mathcal{F}$, where intersections are counted algebraically. Since $P = 6, N = 2$ this algebraic intersection number is $6 - 2 = 4$, and indeed we see that (ignoring the two 1–braids) ∂F is a 4–braid. The only thing which is not completely obvious at this time is how to instruct a computer to find a word in the generators of B_4 which represents ∂F from the 3–component 6–braid $EB(\mathcal{F})$. This will be discussed briefly at the end of Section 4, and in more detail in our paper on the implementation of the algorithm.

3.3 The general case: testing for embeddability

We pass to the question of testing the embeddability of an arbitrary tiled surface $\mathcal{F}$. Since the positive tiled surface $\mathcal{F}'$ is a subsurface of $\mathcal{F}$, and since Lemma 3.1 gives a complete test for the embeddability of $\mathcal{F}'$, it is clear that our general embeddability test must include the cyclic order test of Lemma 3.1 (see (i) of Theorem 3.5, stated below) and a corresponding condition on the cyclic order around the negative vertices. In view of the proof of Theorem 3.4, the

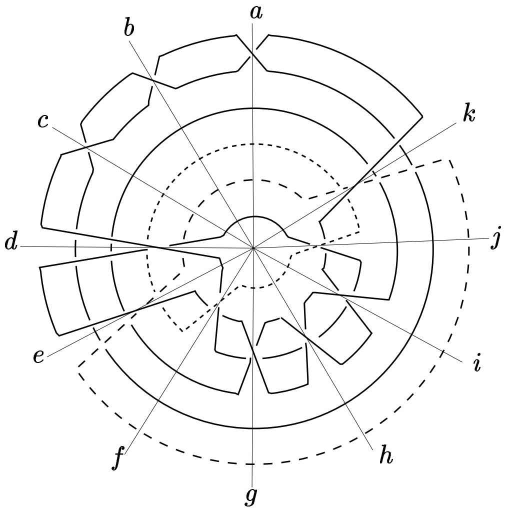


Figure 10: The extended boundary word for the example in Figure 7

remaining obstruction to embedding lies in filling in the disc neighborhoods of the the negative vertices. The obstruction must lie in the b–arcs, which are not present in a positive tiled surface. To describe the obstruction, we need several definitions.

By our hypothesis, the foliation of $\mathcal{F}$ is radial about each vertex. This means that around every vertex there is a leaf which meets the vertex at the angle θ, for every $\theta \in [0, 2\pi]$. Suppose the singular leaves occur at angles $\theta_1, \ldots, \theta_S$. Consider $b(v_i, v_j)$, a b–arc joining vertices v_i and v_j. There is some maximal open interval, (θ_m, θ_n) in which for any $\theta \in (\theta_m, \theta_n)$, there is a b–arc between vertices v_i and v_j which is homotopic to $b(v_i, v_j)$ rel endpoints. Let $[b(v_i, v_j)]$ be the equivalence class given by these b–arcs.

By a slight abuse of notation we say that the b–arc $b(v_i, v_j)$ *exists in the* θ *interval* (θ_{p-1}, θ_p) if $(\theta_{p-1}, \theta_p) \subset (\theta_m, \theta_n)$, ie, if some representative of the equivalence class $[b(v_i, v_j)]$ exists between angles θ_{p-1} and θ_p. Define $gb(v_i, v_j)$ to be a *gb–arc* (or *generalized* b–arc) if either it is a true b–arc, or if v_i and v_j are the positive vertices associated to a single type aa singularity at θ_p. In the latter case, we define the arc $gb(v_i, v_j)$ to lie in then interval (θ_{p-1}, θ_p). Notice that we do *not* include the corresponding arcs for an ab singularity, we will not need them. When we do not need to distinguish between the gb–arcs which are b–arcs and those which are not b–arcs, we will use the simpler notation $v_i v_j$.

Example Table 1 illustrates the table of gb arcs for the example of Figure 7.

interval	gb–arcs		
(θ_1, θ_2)	$b(v_4, v_{2.2})$	$b(v_5, v_{2.1})$	$gb(v_1, v_2)$
(θ_2, θ_3)	$b(v_4, v_{2.2})$	$b(v_5, v_{2.1})$	$gb(v_1, v_2)$
(θ_3, θ_4)	$b(v_4, v_{2.2})$	$b(v_5, v_{2.1})$	$gb(v_1, v_6)$
(θ_4, θ_5)	$b(v_4, v_{2.2})$	$b(v_5, v_{2.1})$	$\cdots$
(θ_5, θ_6)	$b(v_4, v_{2.2})$	$b(v_1, v_{2.1})$	$\cdots$
(θ_6, θ_7)	$b(v_6, v_{2.2})$	$b(v_1, v_{2.1})$	$gb(v_5, v_3)$
(θ_7, θ_8)	$b(v_6, v_{2.2})$	$b(v_1, v_{2.1})$	$gb(v_5, v_3)$
(θ_8, θ_9)	$b(v_6, v_{2.2})$	$b(v_1, v_{2.1})$	$gb(v_5, v_3)$
(θ_9, θ_{10})	$b(v_6, v_{2.2})$	$b(v_1, v_{2.1})$	$\cdots$
$(\theta_{10}, \theta_{11})$	$b(v_4, v_{2.2})$	$b(v_1, v_{2.1})$	$\cdots$
(θ_{11}, θ_1)	$b(v_4, v_{2.2})$	$b(v_5, v_{2.1})$	$gb(v_1, v_2)$

Table 1: The table of gb–arcs for the example in Figure 7

The dotted entries indicate the intervals which end at an ab–singularity; for such a singularity there is no gb–arc which is not a b–arc. In a more complicated example the same would be true for bb–singularities. It is a consequence of our definitions that there are exactly $N = 2$ arcs of type b in every interval and either one or no arcs which have type gb but not type b.

Our embeddability test is given by the following theorem:

Theorem 3.5 *Let $\mathcal{F}$ be a tiled surface whose regions have been labelled in the manner just described. Then $\mathcal{F}$ is embeddable if and only if:*

(i) *The singularities about each positive (respectively negative) vertex are positively (respectively negatively) cyclically ordered in the fibration, with respect to increasing polar angle θ.*

(ii) *The vertices about each positive (respectively negative) singularity are positively (respectively negatively) cyclically ordered on the oriented braid axis, and*

(iii) *The endpoints of a gb–arc in the interval $(i-1, i)$ never separate the endpoints of a b–arc in the same interval.*

Proof We begin the proof by establishing a set of tests which look much more complicated than the tests in Theorem 3.5, but which will turn out to be equivalent to them. Let $\mathcal{F}$ be a tiled surface with singularities at angles $\theta_1, \ldots \theta_S$. We claim that $\mathcal{F}$ is embeddable if and only if it passes the following four tests.

(1) The singularities about each positive (respectively negative) vertex are positively (respectively negatively) cyclically ordered in the fibration.

(2) The vertices about each positive (respectively negative) singularity are positively (respectively negatively) cyclically ordered in the braid axis.

(3) Let vw be a b–arc which exists during the θ–interval $(i-1,i)$. Then $\mathcal{F}$ is not embeddable if the vertices v and w are separated on A by the endpoints of any other b–arc which exists in the θ–interval $(i-1,i)$.

(4) Suppose the singularity at θ_i is type aa, between a–arcs at vertices v and w. Then $\mathcal{F}$ is not embeddable if the vertices v and w are separated on A by the endpoints of one of the b–arcs which exist in the θ–interval $(i-1,i)$.

(5) Suppose the singularity at θ_i is type ab, between an a–arc with vertex endpoint x and a b–arc uv, where u is positive. Then $\mathcal{F}$ is not embeddable if there is different b–arc, say yz, which occurs during $(i-1,i)$ such that x is separated from uv on A by yz.

(6) If the singularity at θ_i is type bb, let u,v,w,x be the vertices of the bb tile T, oriented as they are encountered in traversing ∂T counterclockwise on the positive side of $\mathcal{F}$, with u positive. Then $\mathcal{F}$ is not embeddable if there is a b–arc in the interval $(i-1,i)$ which separates uv from wx.

To prove the claim we first notice that (1) is a necessary condition for the surface to be embeddable, because the foliation is radial in a sufficiently small neighborhood of every vertex. Similarly, (2) is necessary because the singular leaves meeting at a singularity are embedded in a single H_θ. The vertices at the ends of the leaves are all in the braid axis on the boundary of H_θ inducing an order on the vertices. The sign of the singularity indicates whether the surface is locally oriented compatibly with H_θ, or with reversed orientation.

Consider the intersections of the given orientable surface F with the fibers of $\mathcal{H}$, as θ ranges over the interval $[0,2\pi]$. Let N be the number of negative vertices and let P be the number of positive vertices. An Euler-characteristic count shows that there must be P arcs in all, with exactly N of them type b and the remaining ones type a.

We first find necessary conditions for embeddability. If F is embedded, then it has no self-intersections. Since F intersects each non-singular fiber transversally, it follows that a necessary condition for embeddability is that $F \cap H_\theta$, where H_θ is non-singular, be a collection of pairwise disjoint arcs, with N of them of type b and the remaining $P-N$ type a. The b arcs divide H_θ, but the a arcs do not. If there are b–arcs, say $uv, wx \subset H_\theta$, they will intersect if and only if u, v separate w, x on $A = \partial H_\theta$. Let $H_{\theta_1}, \ldots, H_{\theta_s}$ be the singular fibers, in their natural cyclic order in the cycle of fibers around A, with subscripts mod s. If θ varies over the open interval $\theta \in (\theta_{i-1}, \theta_i)$ its intersections with F will be modified by isotopy rel A. Thus a necessary condition for $\mathcal{F}$ to be embeddable is that it pass test (4) for some $\theta \in (\theta_{i-1}, \theta_i)$ for every $i = 1, 2, \ldots, S$.

Next we ask what happens to the intersections of our embedded orientable surface F with H_θ when θ passes through a singular angle in the fibration. There are three types of singularities, ie aa, ab, and bb. It's easy to see that the arcs in the set $F \cap H_\theta$ only change in a manner which can be realized by an isotopy after an aa singularity, but that is not the case after an ab or bb singularity. As we approach the singular fiber which separates the intervals $(i-1, i)$ and $(i, i+1)$ during an aa singularity the a–arcs with endpoints at v, w must approach one-another. But if v and w are separated on A by the endpoints of a b arc which exists during the interval $(i-1, i)$ that will be impossible without a self-intersection in F. The reasons are the same for type ab and bb. Thus, tests (4)–(6) are also necessary conditions for embeddability.

In fact these tests are also sufficient. Assume that all 6 tests have been passed. Then $F \cap H_\theta$ is a collection of pairwise disjoint arcs, with N of them type b and $P - N$ of them type a, for every non-singular fiber. Also, in every singular fiber there is exactly one pair of intersecting arcs, namely the leaves of the associated saddle-point tangency. The union of all of the arcs $F \cap H_\theta$ as θ varies over the closed interval $[0, 2\pi]$ is the trace of the isotopy of $F \cap H_\theta$ as θ varies over $[0, 2\pi]$. The claim is proved.

To complete the proof of Theorem 3.5 we now observe that test (i) of the theorem is identical with (1) of this lemma, and test (ii) is identical with (2). Test (iii) of the theorem is identical with (3) plus (4). It remains to show that tests (5) and (6) are subsumed by test (iii).

The key fact to notice is that the changes as we pass through an ab (respectively bb) singularity at the angle θ_i involve exactly one (respectively two) b arc (respectively arcs). All other b–arcs in the interval $(i, i+1)$ are identical with those in the interval $(i-1, i)$.

Consider test (6) first. Suppose that there is a bb singularity at θ_i, as in (6), with b arcs uv and wx in $(i-1, i)$, and new b–arcs vw and ux in $(i, i+1)$. Suppose also that there is a b–arc cd in the interval $(i-1, i)$ which separates uv and wx. Then c and d separate u and v. However, in the interval $(i, i+1)$ (ie after the singularity) there will be new b arcs vw, ux. The b–arc cd will still be present. But that is impossible by (3), because c and d separate u and v.

Consider test (5) next. Suppose that there is an ab singularity at θ_i, as in (5), such that the b–arc uv is in $(i-1,i)$ and the b–arc xv is in $(i,i+i)$. All other b–arcs in $(i-1,i)$ are also in $(i,i+1)$. Suppose that x is separated from the b–arc uv by some other b–arc yz in the interval $(i-1,i)$. By (3), the arc yz cannot cross uv. This means that y and z separate x from both u and v. Passing to the interval $(i,i+1)$, the arc yz is still present. However yz crosses xv. But that is also impossible, by (3). The proof of Theorem 3.5 is complete. □

Example We illustrate the embeddability test on the example which was given in Figure 7, using the data in Table 1. Recall that the direction of increasing polar angle θ is counterclockwise (respectively clockwise) about a positive (respectively negative) vertex. An easy check shows that the order is correct about every vertex, so the example passes test (i) of Theorem 3.5. Similarly, test (ii) is passed. We turn to test (iii). There are two negative vertices, and so there are two b–arcs in each non-singular fiber. Inspecting Figure 7, we see that there is a b–arc joining vertices v_4 and $v_{2.2}$ in the interval (θ_{10},θ_6), and one joining vertices v_6 and $v_{2.2}$ in the interval (θ_6,θ_{10}). These are the entries in the first column of Table 1. Similarly for the b–arcs which end at vertex $v_{2.2}$, which are recorded in the second column. As for the gb–arcs, we see that there are aa–singularities at $\theta_1,\theta_2,\theta_3,\theta_4,\theta_7,\theta_8,\theta_9$, explaining the entries in the third column of Table 1. Inspecting the rows of the table, one at a time, we verify that the endpoints of a gb–arc (remembering that gb–arcs include b–arcs) never separate the endpoints of a b–arc. Thus test (iii) of Theorem 3.5 is also passed.

3.4 Eliminating inessential tiled surfaces

For efficiency, we will want to be able to eliminate inessential tiled surfaces as we do the enumeration. The following proposition will allow us to do so.

Proposition 3.6 *Let $\mathcal{F}=(F,G,C)$ be a tiled surface which passes the embeddability test of Theorem 3.5. Then $\mathcal{F}$ is essential if, for every b–arc β in the tiled surface, the two points in $\partial\beta$ are not adjacent in the cyclic ordering of the vertices on A.*

Proof Let β be a b–arc in $F \cap H_\theta$ for some non-singular fiber H_θ. Recall that β divides divides H_θ into two subdiscs, δ_1 and δ_2, and that β is *inessential* if one of these discs, say δ_1, has empty intersection with K. The subdisc δ_1 may contain other b–arc within it, but we may assume that β is an innermost b–arc and δ_1 contains no other b–arcs.

In this case, we can simplify the braid foliation of $\mathcal{F}$ by pushing F through a neighborhood of δ_1 in 3–space to eliminate two points of $A \cap F$. We can detect this situation combinatorially because, if β is inessential and innermost, its two endpoints will be adjacent vertices on A. $\square$

4 Enumerating closed braid representatives of the unknot

Our task in this section is develop a procedure for enumerating the closed n–braid representatives of the unknot, up to conjugacy, for each fixed n. From now on we restrict our attention to the case when the F is a disc. To stress this, we use the notation $\mathcal{D} = (D, G, C)$. Each closed n–braid representative of the unknot is the boundary of a embeddable tiled disc, and our plan is to enumerate all embeddable tiled discs and read off their boundary words. If $n \geq 4$ there will be infinitely many conjugacy classes. Our measure of complexity for the enumeration is the number of vertices in the embeddable tiled disc, ie the number of points in $A \cap D$.

We refer the reader to Figure 11. The top left sketch is a closed 4–braid diagram for the unknot. It is readily seen to be the unknot, however the disc that it bounds is a little obscure. The top right sketch illustrates the same closed braid from a different angle. The disc meets the braid axis A in 8 points (the vertices of the induced foliation), labelled $1, 2, 3, 3.1, 3.2, 4, 5, 6$. The two labelled 3.1 and 3.2 are negative vertices. The singularities are not labelled, but there are 7 of them. (By an Euler characteristic argument, there is always exactly one fewer singularity than vertex.) Singularities occur on each of the three narrow twisted bands, and each of the narrow, vertical tubes coming up out of discs 1 and 2 each have two singularities of opposite signs. Vertices are labelled with numbers, singularities with letters. Both singularities and vertices have a label of plus or minus, too. There is an embedding of a model disc $\mathcal{D}$in 3–space which realizes this geometry.

The bottom sketch in Figure 11 shows the model disc. The graph of the singular leaves has been pulled back to the model disc, and decorated to show the

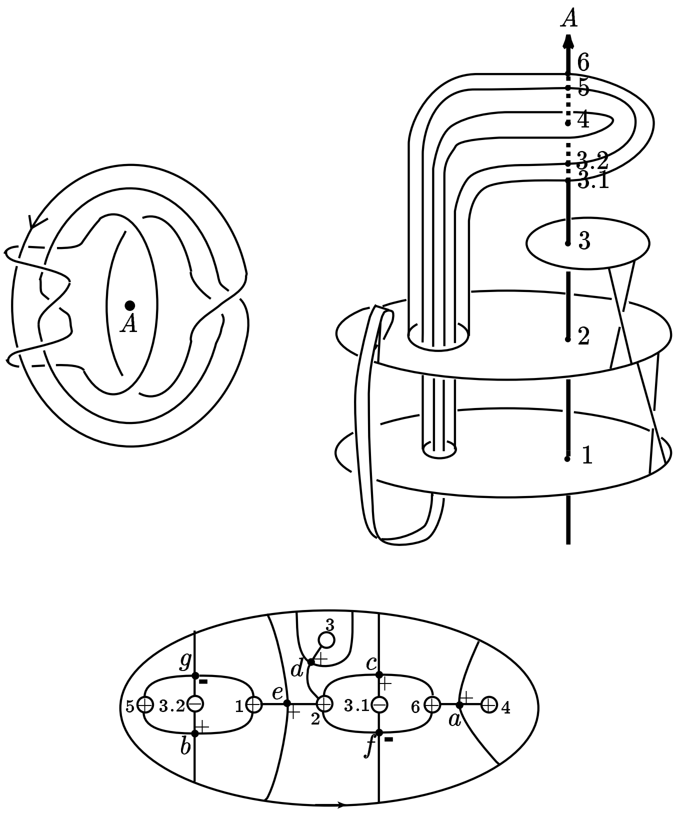


Figure 11: Several views of a closed braid which bounds an interesting disc

order and signs of the vertices and singularities. This bottom sketch is our embeddable tiled disc. The problem which we address now is this: let us suppose that we were handed the 4–braid example K which is illustrated in the left sketch in Figure 11, and we want to verify algorithmically that it is the unknot. Our plan is to enumerate a suitably long list of foliated discs whose boundaries are 4–braids, and to check our given example K against the members of the list. So we need to learn how to generate, systematically, a list of embeddable tiled discs all of whose boundaries have braid index 4, which is long enough to contain the example in Figure 11.

In view of Lemma 3.3 our plan is to fix $n = P - N$ and to enumerate embeddable tiled discs in order of increasing v. This is the same as enumerating embeddable tiled discs with $(P, N) = (n, 0), (n+1, 1), (n+2, 2), \ldots$ in that order.

To enumerate the embeddable tiled discs we apply ideas first used in the proof of Lemma 1 of [1]. Again we refer the reader to [3] for details, and give a brief summary here. To do so we introduce a move which is guided by the foliation of the surface and allows us to change an arbitrary embeddable tiled disc to a new embeddable tiled disc. This new embeddable tiled disc is simpler in the sense that it has fewer negative vertices. Our algorithm will attempt to reverse this process, starting with a simple embeddable tiled disc and generating more complex ones.

Definition *Stabilization along an ab–singularity* Recall that, by hypothesis, the foliation is radial is a neighborhood of each vertex in the tiling. The top row in Figure 12 shows how, any time there is an ab–singularity in the foliation, we may push K across the singularity and its associated negative vertex, in a neighborhood of the separating leaf which meets K, to a new position which is again everywhere transverse to the foliation. It follows that after we do this move we will have a new closed braid representative, say $K^\star$, of the unknot. Notice that after stabilizing, a bb singularity may have become an ab singularity. The middle row of pictures shows why the move *increases* the braid index from n to $n+1$, while *decreasing* the number N of negative vertices from N to $N-1$. The bottom row shows our stabilization move on the embedded surface in 3–space. If one looks carefully one can see the half-twist which has been introduced in the course of the push. We note that the pictures of ab–tiles in the bottom row of Figure 12 are deformations of the picture in Figure 5: we stretched out the top sheet to make visible a neighborhood of the singular leaf.

Theorem 4.1 *Let $\mathcal{D}$ be an arbitrary embeddable tiled disc. Suppose that the graph of $\mathcal{D}$ contains P positive vertices and N negative vertices. Then there exists a sequence of embeddable tiled discs:*

$$\mathcal{D} = \mathcal{D}_0 \to \mathcal{D}_1 \to \ldots \to \mathcal{D}_{N,}$$

where each $\mathcal{D}_{i+1}$ is obtained from $\mathcal{D}_i$ by a single ab–stabilization, so $\mathcal{D}_N$ has only aa–singularities. If the initial the $\mathcal{D}$is essential, then so is each $\mathcal{D}_i$.

Equivalently, every embeddable tiled disc with P positive vertices and N negative vertices may be constructed by finding a embeddable tiled disc the graph of which is a tree of P positive vertices, then adding N ab–tiles, one at a time, to the graph. At each addition stage, the new vertex and singularity are inserted into the orders of the older vertices and singularities and in such a way that the new graph corresponds to an embeddable tiled disc. If the disc to be constructed is essential, then each intermediate disc may also be assumed to be essential.

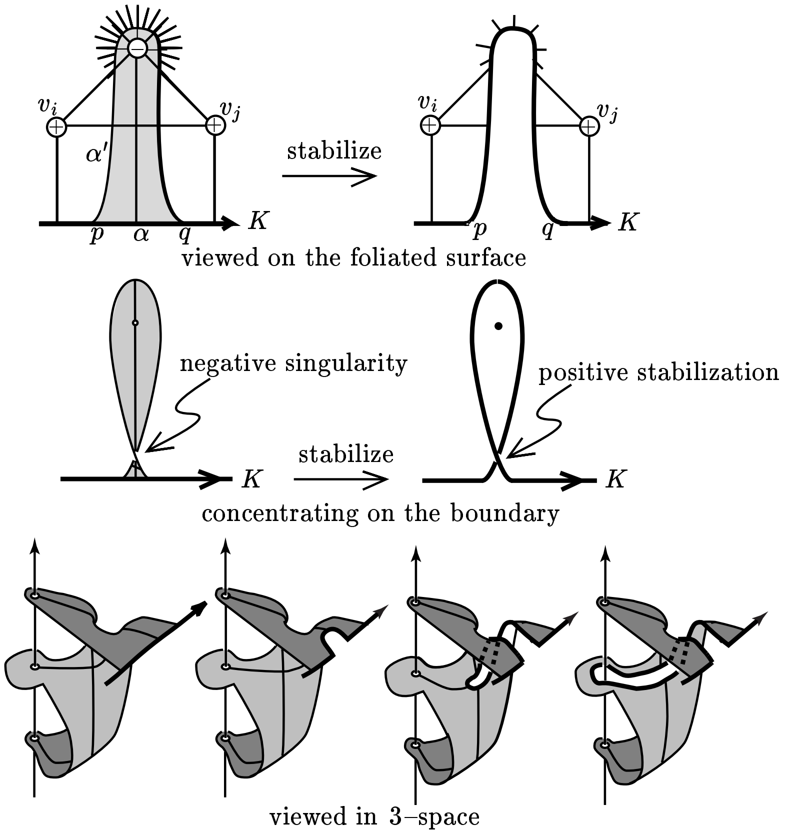


Figure 12: Stabilization along an ab–tile, viewed from three perspectives

Proof We begin with the given embeddable tiled disc $\mathcal{D}$, which, by hypothesis, contains N negative vertices. If $N = 0$ we are done, so assume that $N > 0$. From Figure 5 we can see that the foliation of $\mathcal{D}_0$ necessarily contains singularities of type bb or ab, because singularities of type aa only connect to positive vertices. But if there are singularities of type bb, then there must also be singularities of type ab because a bb tile can only be glued to another bb tile or to an ab tile. However, a subsurface of D cannot be composed entirely of bb–tiles, for if it were it would be closed, and also entirely in the interior of D, which is absurd. So we may assume that there is at least one ab–singularity. It is then possible to stabilize along the ab singularities, one at a time, as in Figure 12 until we obtain a tiled disc $\mathcal{D}_N$ which has no negative vertices.

We must show that each $\mathcal{D}_i$ is embeddable and has no inessential b–arcs. Since the graph of $\mathcal{D}_N$ has no negative vertices, its singularities must all be type aa.

Since $\mathcal{D}$is embeddable, and $\mathcal{D}_i \subset \mathcal{D}$ for all i, it follows that each $\mathcal{D}_i$ is an embeddable tiled disc. The b–arcs of each $\mathcal{D}_i$ are also b–arcs of $\mathcal{D}$. Consider a b–arc of $\mathcal{D}_i$. It is isotopic to one inside a non-singular H_θ which we may assume is sufficiently far from any singular H_ϕ. Since ab–stabilization occurs in a neighborhood of a singularity which is inside a singular H_ϕ,

$$\partial D \cap H_\theta \subset \partial D_i \cap H_\theta,$$

and the b–arc is essential in D if and only if it is essential in D_i.

Thus, the sequence of tiled discs is a sequence of (essential) embeddable tiled discs. □

In the previous theorem we examined how arbitrary essential tiled discs can be simplified by stabilization along ab–singularities. We showed that after some number of such stabilizations one arrives at a positive tiled disc. The latter contains only positive vertices, and so has only aa–singularities. Its graph of singular leaves is a tree. As was shown by Bennequin in [1], the new tiled disc can then be further simplified to the trivially tiled disc by removing aa–singularities which belong to vertices of valence 1, one at a time. That process is known as *destabilization along aa–singularities*. See Figure 13.

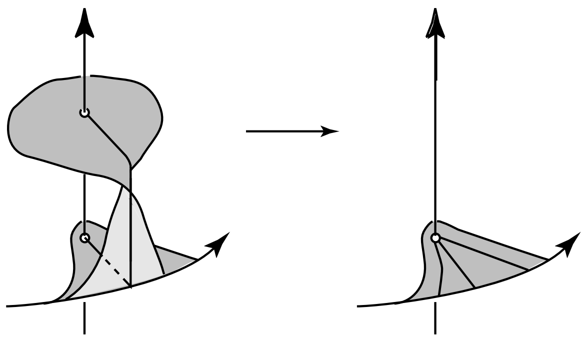


Figure 13: Destabilization along an aa–tile having a vertex of valence 1

See [3] for a full discussion of how the two processes, ie a finite number of stabilization along ab–singularities followed by a finite number of destabilizations along aa–singularities can be used to change an arbitrary closed n–braid representative of the unknot to an $(n+m)$–braid representative which bounds a simpler tiled disc and thence to the trivial 1–braid representative. We now consider the reverse of stabilization along ab–singularities. This reverse proceess will be used to build up all tiled discs.

Theorem 4.2 *All possible embeddable tiled discs of fixed braid index n may be enumerated in order of increasing v by the following (not necessarily efficient) procedure:*

- *Enumerate all positive tiled discs with n vertices, testing each for embeddability and discarding non-embeddable examples. This gives a list of all possible embeddable tiled discs of braid index n with $v = n$ vertices.*
- *To enumerate all embeddable tiled discs with $n+2$ vertices, first enumerate all positive tiled discs with $n+1$ vertices, testing each for embeddability and discarding non-embeddable examples. Then add one ab tile, using the reverse of stabilization along an ab–singularity, in all possible ways, testing every position for embeddability and essentiality and discarding non-embeddable or inessential examples. This produces a list of all possible essential embeddable tiled discs of braid index n with $v = n+2$ vertices.*
- *To enumerate all embeddable tiled discs with $n+4$ vertices, first enumerate all positive tiled discs with $n+2$ vertices, then discard any non-embeddable examples. Then add two ab tiles in all possible ways, and discard any non-embeddable or inessential examples. This produces a list of all possible embeddable tiled discs of braid index n with $v = n+4$ vertices.*
- *Continue in this way for $v = n+6, n+8, \ldots$. For $v = n+2i$, the enumeration begins with all possible positive embeddable tiled discs with $n+i$ vertices, and continues by adding i tiles of type ab.*

Proof By Lemma 3.3 we know that $n = P-N$. Since $v = P+N$, the enumeration begins with $(P,N) = (n,0)$ and continues with $(n+1,1), (n+2,2), \ldots$. By Theorem 4.1 every essential embeddable tiled disc will eventually appear on the list. A key fact in the enumeration is that an embeddable tiled disc can never be obtained from a non-embeddable tiled disc by the reverse of stabilization along an ab–singularity. The reason is that each $\mathcal{D}_i$ in any sequence of tiled discs constructed by repeated ab–stabilization on a embeddable tiled disc is embeddable. Similarly, by Lemma 4.1 an embeddable tiled disc with essential b–arcs can never be obtained from a embeddable tiled disc having any inessential b–arcs by the reverse of stabilization along an ab–singularity. □

Corollary 4.3 *All possible conjugacy classes of n–braid representatives of the unknot may be enumerated in order of complexity (n,v) by enumerating all embeddable tiled discs, using Theorem 4.2, and then using Theorem 3.4 to determine the associated boundary words.* □

Example We illustrate the enumeration process for the example which we considered earlier, in Figure 11. See Figure 14.

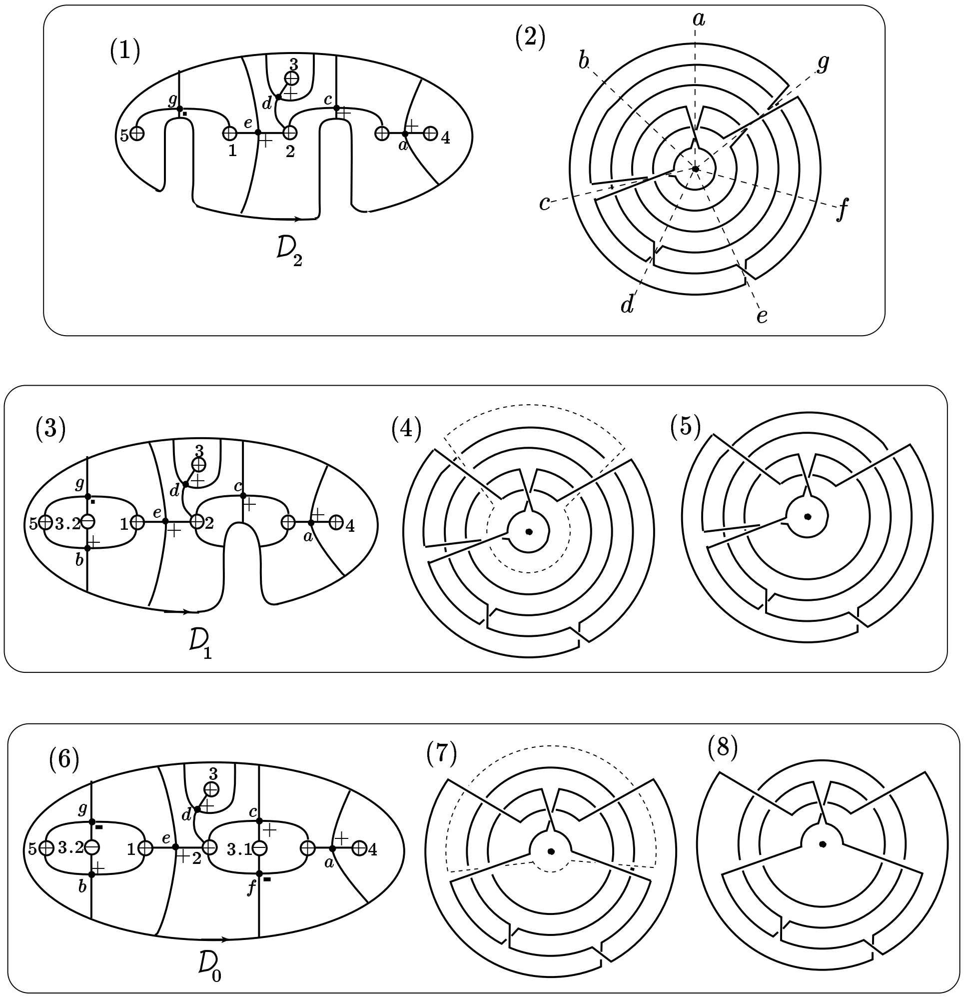


Figure 14: Working through the details in the example of Figure 11

Sketches (6),(3),(1) of Figure 14 are the successive modifications of the embeddable tiled disc which we first met in Figure 7. Call the three tiled discs $\mathcal{D}_0, \mathcal{D}_1$ and $\mathcal{D}_2$, respectively. The initial embeddable tiled disc $\mathcal{D}_0$ has 8 vertices (at heights 1, 2, 3, 3.1, 3.2, 4, 5, and 6) and 7 singularities, labelled a, b, c, d, e, f, g. Alternatively, our enumeration process begins with the positive embeddable tiled disc $\mathcal{D}_2$ in sketch (1). Its embedding is illustrated in sketch 2. It is made of discs connected by twisted bands. The discs are $\delta_1, \delta_2, \ldots, \delta_6$

of heights $1, 2, \ldots, 6$ and radii $6, 5, \ldots 1$, respectively. Its boundary word is $a_{6,4}a_{6,2}a_{3,2}a_{2,1}a_{5,1}^{-1}$.

In sketch (3) we have added the ab–tile associated to singularity b to make $\mathcal{D}_1$. We could equally well have added singularity f first, it would not matter. The reader may find it interesting (we did!) to try adding f first, then b, and to see the pleasing way in which the intermediate braids change, while both choices lead to the same final braid (sketch (8)).

The component of $EB(\mathcal{D}_1)$ that is not in ∂D_1 is lightly dashed in sketch (4). The new boundary of ∂D_1 (ie, after deleting the dashed circle) is shown in sketch (5).

The reader may have wondered about the negative vertex at level 3.2. As an aside, we now suggest that the interested reader start with the embedded disc of sketch (2) and add yet one more disc $\delta_{3.2}$ of radius $= 6 - 3.2$, at a level between δ_3 and δ_4, only now with its negative side facing 'up'. The boundary of $\delta_{3.2}$ can be seen to be homotopic through an embedded annulus to the dashed unknot in sketch (4) of Figure 14. Let us refer to the disc with the annulus attached as the new disc.

The new disc meets the old surface along part of its boundary—all of the dashed unknot of sketch (4) except the dashed arc at polar angle (b). Further deform the new disc slightly to take the dashed arc to the solid arc at angle (b) in sketch (4). If we now form the union of the old surface and new disc, behold, we have a new embedded surface whose boundary is exactly that of sketch (5)! It's wonderful to see how the pasting can be done without introducing any intersections with the previous surface.

In this way we obtain an embedded disc D_1 which is bounded by the braid in sketch (5). The piece of the old boundary which was on the dashed circle and away from the wedge of 3–space near polar angle b is in the interior of D_1, and the half-band at polar angle b is a subarc of the new boundary.

Sketches (7) and (8) show the next stage of adding another ab–tile at singularity f to get $\mathcal{D}_0$. Again the new unknot component is lightly dashed. Again there is a missing disc $\delta_{3.1}$, and it fits right between δ_3 and $\delta_{3.2}$ to give the embedded surface $\mathcal{D}_0$, which is exactly the surface in Figure 11, constructed in a systematic way.

The braid in sketch (8) is ∂D_0. It is clearly a 4–braid, though there is some question as to how one can read off the braid word efficiently and algorithmically because its strands skip between 6 different discs. There are many methods for this which essentially involve keeping a table of which strands are in the braid component at each singular angle. We skip the details here. More information will be provided in a future paper on implementation details of the algorithm.

The list of embeddable tiled discs which we have just constructed has duplications. Some redundancy is caused by non-uniqueness of foliations; we will obtain duplicate embeddable tiled discs which differ only by changes in foliation, which at most change the boundary by an isotopy in the complement of the axis. Additional redundancies occur because non-isotopic embeddable tiled discs can have boundaries which represent the same conjugacy class in B_n. In another paper we will consider the problem of implementing the algorithm, and at that time we will address these issues.

Remark 4.4 The reader who has followed the details of the calculation which we illustrated in Figure 14 will have learned that the boundary of the embeddable tiled disc which is illustrated in sketch (6) of that Figure, and also in the bottom sketch in Figure 11, is represented by the 4–braid:

$$a_{4,2}a_{3,2}a_{2,1}a_{4,3}a_{3,2}^{2}a_{2,1}a_{3,2}^{-1}a_{4,3}^{-1}a_{2,1}^{-1}a_{3,2}^{-1}.$$

On the other hand, the closed braid which is illustrated in the top left sketch in Figure 11 is represented by the 4–braid:

$$a_{4,3}a_{3,2}^{-1}a_{2,1}^{-2}a_{3,2}^{-1}a_{3,2}a_{3,2}a_{2,1}^{2}a_{3,2}a_{4,3}^{-1}a_{3,2}a_{2,1}.$$

A direct attempt to show that these two 4–braids are conjugate in B_4 will very likely lead to frustration. Fortunately, an algorithmic solution to this difficult problem is available. Indeed, we have interfaced it with our program for the algorithm. We refer the reader to Appendix B, where it is described, briefly.

5 The halting theorem

In the previous sections we have shown that there is an algorithm which enumerates all foliated discs. By reading the braid word of the boundary of each disc we then get a list of all possible (conjugacy classes of) unknots. The problem which remains is to learn when to stop testing and conclude that the knot in question is not going to appear on the list and, hence, is truly knotted.

The boundary of a foliated disc is a closed braid, but our given knot K will in general not be a closed braid. The first step is to change K to a closed braid. There are many ways to do this. If the knot has n Seifert circles and k_0 crossings, the method given in Appendix A is simple and it converts K to a closed braid of n strands and word length $k \leq k_0 + (n)(n-1)$. The pair

(n,k) is then a measure of complexity of the given example. But (n,v) is our measure of complexity of the foliated disc. Our main task in this section is to find a relationship between (n,v) and (n,k). Clearly, $n=n$, and we will show that there is a relationship of the form $v<f(n,k)$ where f is an appropriate function. This will prove that we need not look for arbitrarily complicated discs.

Establishing such an upper bound on v using foliated surface techniques is an interesting and significant open problem—one we have not solved. We believe that a solution which is in the spirit of our algorithm exists, but that finding it will depend upon obtaining a better understanding than we have at this time of the boundary words. We have been able to prove that an upper bound exists by using the machinery in [5], but we were unable to find an explicit formula. We have not included that proof because it requires the development of a great deal of background material.

We present a different approach. We establish our upper bound by first constructing a triangulation of the complement of K which is adapted to closed braids. In particular, we do it so that the braid axis meets α tetrahedra, and meets them in a controlled way. We will need to count the total number t of tetrahedra in the triangulation. After that we will use the Kneser–Haken theory of normal surfaces to obtain an upper bound, depending on t, on the number of intersections of the axis with a single tetrahedron. Multiplying by α we will obtain the bound that we need.

5.1 Constructing the triangulation

Our goal in this section is to construct the triangulation and to prove Lemma 5.1 below. A suitable triangulation of S^3 is one for which:

(1) The knot K is in the 1–skeleton,

(2) A regular neighborhood of K is triangulated

(3) The knot K is a closed braid, and the braid axis meets a fixed number α of tetrahedra, in a fixed way, ie as a straight line from the center of one of the faces to the opposite vertex.

The construction of a suitable triangulation of S^3 involves many technical details. The reader who is willing to accept the fact that we can construct one with the stated properties will not lose the thread of the argument by proceeding at this time directly to Lemma 5.1, and accepting its truth.

In the interests of efficiency of the algorithm, we will try to minimize t and α in our construction of a suitable triangulation. We start by defining a few pieces of the construction. At several points in the construction we must subdivide. We need several specific subdivisions to do this well.

Definition Each crossing of the braid strands will be carried by a tetrahedron which we subdivide with the *crossing triangulation* which is defined as follows. Given a tetrahedron T with vertices A, B, C, D as in Figure 15(a), start by slicing it by two planes P_1, P_2 which separate the tetrahedron into 3 pieces as in Figure 15(b). The top and bottom pieces are affine triangular prisms and contain $\overline{AB}$ and $\overline{CD}$ respectively. The middle piece is an affine cube which can be further divided into two affine prisms.

An affine prism can be triangulated with 3 tetrahedra. Consider the prism $ABEFGH$ in Figure 15(c). The simplices $ABEG, BEFG$ and $BFGH$ are a triangulation of the prism. After cutting the middle piece into two prisms, T has been divided into 4 prisms, each of which we can subdivide into 3 tetrahedra in a compatible way to make a triangulation of $ABCD$ with 12 tetrahedra. We call this subdivision the *crossing triangulation*. Note that the crossing triangulation of T has 5 triangles in each original face of T.

Definition Given a tetrahedron, consider the *frustum* obtained by slicing off one corner (say, the top corner) with a plane (see Figure 16). This frustum is a convex polyhedron with 3 quadrilateral faces and two triangular faces. The *frustum triangulation* is the triangulation achieved by drawing in either of the two diagonals in each quadrilateral face, thus triangulating the boundary, then taking the cone of the triangulation on the boundary into any interior point. The boundary of the triangulated frustum has 8 faces, so there are 8 tetrahedra after subdividing. The original tetrahedron has been subdivided into 9 tetrahedra.

Definition The final subdivision of a tetrahedron we will define we call the *edge triangulation* because we will use it for tetrahedra at the edges of our picture. Suppose we have a rectangular pyramid triangulated as in Figure 17(a). Figure 17 shows a pyramid divided into 4 tetrahedra, but in general there might be any finite number. Pick a point D in the edge $\overline{CP}$ and consider the plane containing points A, B and D. Let $D_1, \ldots, D_n$ be the points of intersection between this plane and the edges $\overline{C_1, P}, \ldots, \overline{C_n, P}$. This plane cuts each of the original tetrahedra into 2 pieces, one of which (the one containing P) is a tetrahedron and one which is not a tetrahedron.

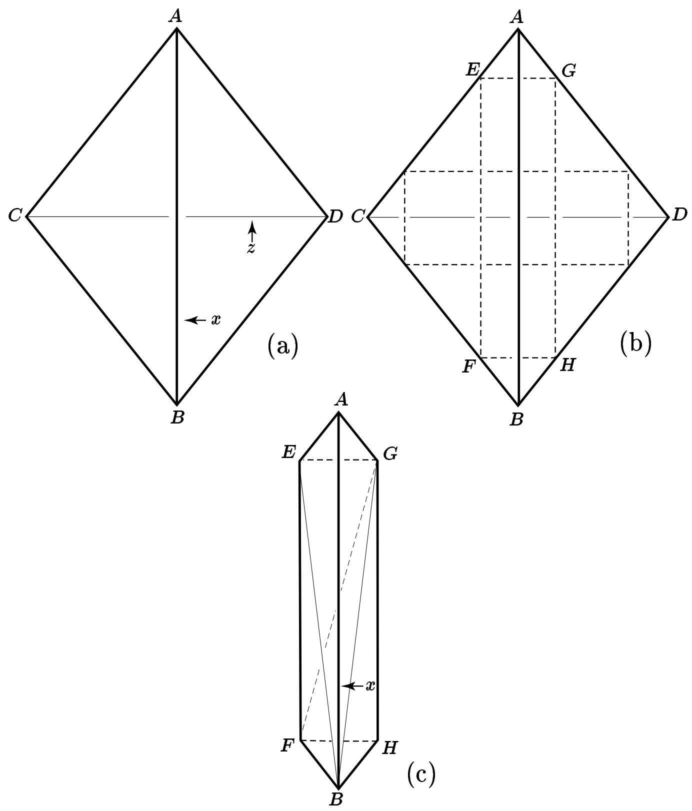


Figure 15: The crossing triangulation

The final step in the construction is to divide the objects which are not tetrahedra by another plane. Consider the plane containing the triangle $AC_kD_{k-1}(0 < k \leq n)$. This cuts the object with vertices $A, C_k, D_k, D_{k-1}, C_{k-1}$ into two tetrahedra. The end result is that each of the original simplices in Figure 17(a) is now 3 simplices in (b), except for the tetrahedron ABC_nP has been divided into 2. Note that the face ABP is not subdivided by this triangulation.

We are now prepared to triangulate S^3 so as to satisfy conditions 1 and 2 from above. Consider Figure 18. It is a triangulated rectangle, but after identification of the top and bottom edges it becomes a triangulated cylinder or annulus. The bold lines represent the original knot (as a braid) and the thinner lines are other edges in the triangulation. The cylinder has been draw with the middle section shown larger than the other two only because that is where the interesting detail lies. It should be understood that the top and bottom portions will be larger when the cylinder is embedded.

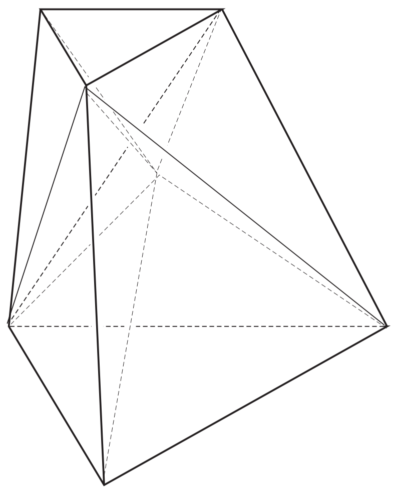


Figure 16: The frustum triangulation

The goal of our triangulation is to minimize t and α while satisfying conditions 1,2 and 3. Satisfying 1 is easy—we build our triangulation by starting with K as out 1–skeleton and extending it. To satisfy 2, we will use the following condition:

Neighborhood Condition

- *No two simplices which meet different strands of K may intersect.*

Clearly, any triangulation satisfying the Neighborhood Condition has a triangulated regular neighborhood of K.

We assume that we have been presented with K as a braid with n strands and word length k in the Artin generators. (See Appendix B for an exposition on how to convert a knot to a braid efficiently.) We think of K as lying on a cylinder winding around and parallel to the braid axis A, where A is the z–axis in 3–space. Furthermore, think of this cylinder as having a triangular cross section, ie, it is made of three flat sides. All the crossings are assumed to be on one side the cylinder, the other two sides carry the closing of the braid into a knot. Thus, in Figure 18 all the crossings of the bold lines are in the middle portion of the figure and in the top and bottom sections there are no crossings.

We start by considering a braid with n strands and no crossings. In this case all three sections will be triangulated like the upper and lower ones. To guarantee

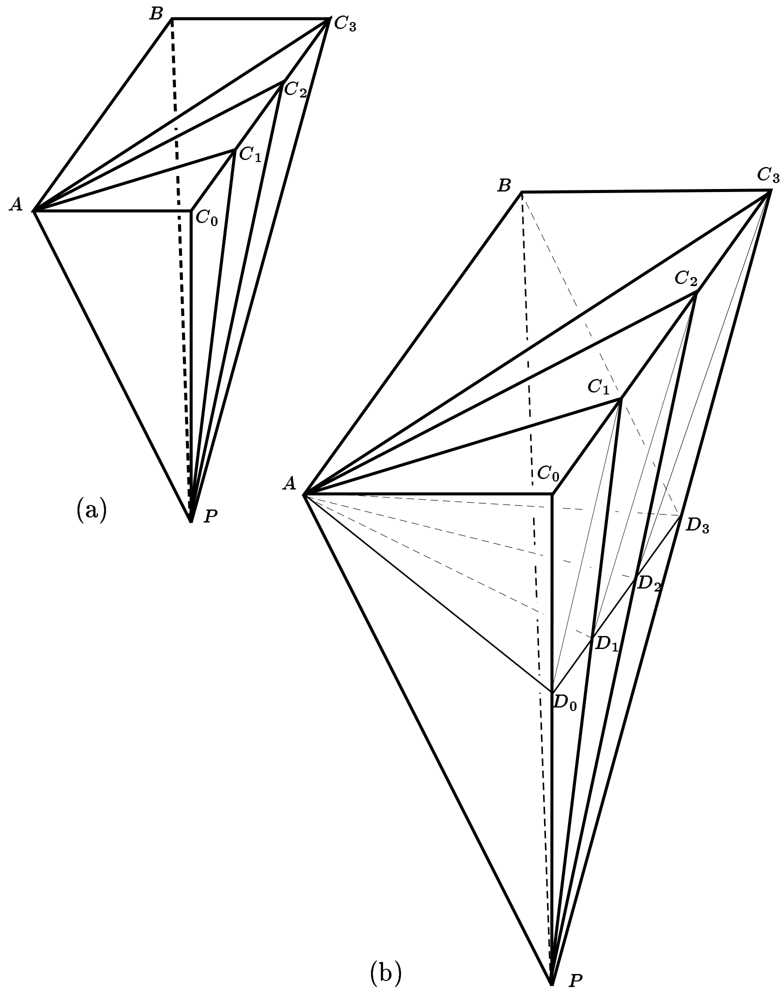


Figure 17: The edge triangulation

the Neighborhood Condition, subdivide the rectangle between each adjacent pair of strands in each section of the cylinder into 3 rectangles, the subdivide each these into a pair of triangles. Thus, for each adjacent pair of strands there are 18 triangles, or $18(n-1)$ in all.

Strands 1 and n are also adjacent to 3 rectangles which we divide into 2 triangle each for 12 more triangles. These are the *edge triangles; the others are* interior triangles. Thus, if K is an n stranded braid with no crossings it lies on a triangulated cylinder with $18n-6$ triangles.

edge
triangles

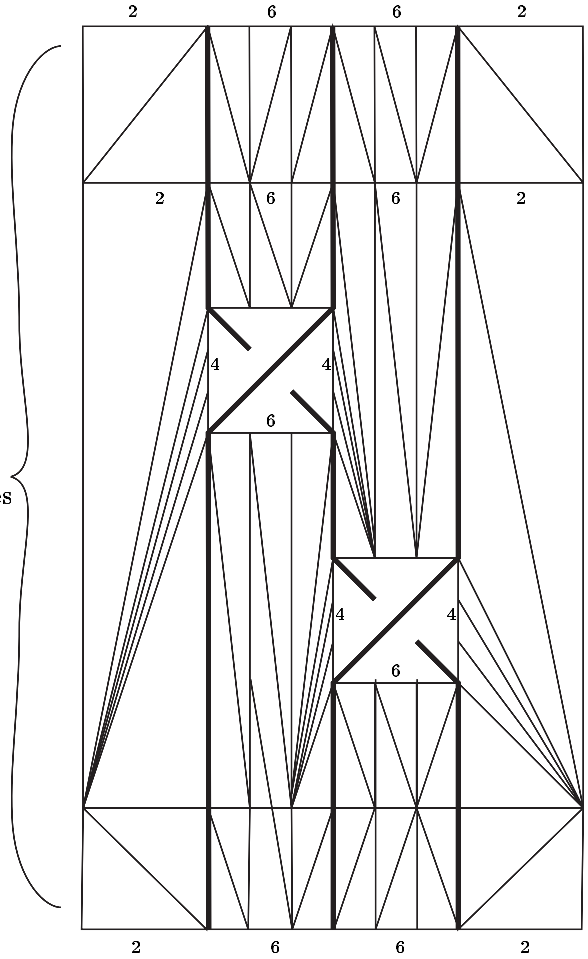


Figure 18: The triangulated cylinder

Now consider adding a crossing. The strands cannot cross and stay on the cylinder, so we fatten the cylinder at each crossing by adding a tetrahedron so that the two strands are on opposite edges of the tetrahedron. We triangulate the tetrahedron with the crossing triangulation described earlier (Figure 15, thus guaranteeing the neighborhood condition. We do not draw this crossing triangulation in Figure 18, but note that it introduces two new vertices on four sides of a square around the crossing. Each crossing breaks the rectangles between the strands into two rectangles (plus the crossing). We subdivide each rectangle as before, converting each triangulated rectangle into 2, introducing 6 more triangles.

The square around the crossing has 4 vertices on its sides, as well. These vertices are in the middle of a vertical edge on a triangle inside a rectangle triangulation. We retriangulate that triangle by adding straight lines from the new vertices to the opposite vertex of the triangle, thus introducing 4 more triangles on each side.

The crossing triangulation has 5 triangles on each face of the tetrahedron, and two faces are on each side of the cylinder. Thus each crossing introduces $6 + 2(4) + 2(5) = 24$ new triangles onto the cylinder. (In addition, each crossing triangulation has 12 tetrahedra which we will count later.) After adding all k crossings we will have a triangulated cylinder with $18n-6+24k$ triangles. If K is a knot, strands 1 and n must be in at least one crossing each, introducing 4 additional edge triangles on each side, so at least 18 triangles are edge triangles and at most $18n + 24k - 24$ are interior triangles.

The boundary of the triangular cylinder is a pair of triangles. Cap off the cylinder with two triangles to make it a sphere triangulated with $18n + 24k - 4$ triangles. Pick two points in A, one enclosed by the sphere and one in its exterior, called the South Pole (SP) and North Pole (NP), respectively.

The cone of the triangulation on the sphere to SP is a triangulated 3–ball with $18n + 24k - 4$ tetrahedra. Unfortunately, the neighborhood condition is now violated because all the tetrahedra meet at a single point, SP. We subdivide each tetrahedron coming from an interior triangle by cutting the "tip" off the tetrahedron, yielding a tetrahedron at SP and an untriangulated frustum. We triangulate each frustum into 8 tetrahedra by the frustum triangulation given earlier, for a total of 9 new tetrahedron for each original one. Thus we get at most $9(18n + 24k - 24)$ tetrahedra from interior triangles.

The tetrahedra coming from the edge triangles we triangulate with the edge triangulation, replacing each edge tetrahedron with 2 or three new tetrahedra. Thus we have at most $9(18n + 24k - 24)) + 3 \cdot 18$ tetrahedra coming from the cylinder. We do this edge triangulation so that the sides of the edge tetrahedra containing the boundary of the cylinder at not subdivided at all.

Finally, the tetrahedra made by coning the the boundary of the cylinder do not need any subdividing. These tetrahedra do not touch the braid, thus they vacuously satisfy the neighborhood condition. Furthermore, the only other tetrahedra touching one of these tetrahedra are edge tetrahedra, and these are not subdivided on the side touching the tetrahedra in question, so there is no need to subdivide them at all.

Thus, our ball has $9(18n + 24k - 24) + 3 \cdot 18 + 2$ tetrahedra. Perform the same construction on the exterior of the sphere to yield twice that many tetrahedra.

Now we add in all $12k$ tetrahedra in the crossing triangulations at each crossing to get $324n+432k-376$ tetrahedra in our triangulated triangulated S^3, which, by construction, satisfies the neighborhood condition and hence condition 1 and 2 from the beginning of this section. Notice that the braid axis A intersects exactly 4 tetrahedra in the triangulation, and does so in a canonical fashion. The reason is that A is the z–axis and passes through SP, NP and the four tetrahedra made by coning the boundary of the cylinder. By construction, it meets each of these tetrahedra as a line segment which runs from the center of one of the faces to the opposite vertex. Thus we have proved:

Lemma 5.1 *Given a closed braid K of n strands and word length k in the Artin generators, there is a triangulation of S^3 satisfying the neighborhood condition with $324n+432k-376$ tetrahedra, only 4 of which meet the braid axis in their interiors.* □

5.2 An upper bound on v

The next step in the formulation and proof of the halting theorem is to obtain an upper bound for how many times the braid axis meets a single tetrahedron in the triangulation. For this, we turn to the theory of normal surfaces in 3–manifolds. A normal surface in a triangulated 3–manifold is a PL surface in the complement of the 0–skeleton with some restrictions on how it can intersect a tetrahedron of the triangulation.

See [9] for a review of the essential facts which we need about normal surfaces. Among them is the fact that the disc which our unknot bounds can be isotoped to a normal surface. Also, each component of the intersection of that normal surface with a tetrahedron is a planar triangle or quadrilateral with corners in the interiors of the distinct edges of the tetrahedron. Also, for each tetrahedron there are 7 different combinatorial types of possible intersections.

Lemma 5.2 *Let M be the closure of S^3 minus the triangulated regular neighborhood of an unknot K. Let t be the number of 3–simplices in the triangulation of M and let S be any simplex in the triangulation of M. Then some longitude of K on ∂M bounds a disc F which is a normal surface, and the number of components of $S \cap F$ is bounded above by $7t2^{7t+2}$.*

Proof Fortunately we do not need to do any work at all. Lemma 6.1 of [10] uses a very different triangulation of a knot complement from ours, but provides exactly the estimate we need: that the number of components of a given combinatorial type in $S \cap F$ is at most $t2^{7t+2}$. We thank Joel Hass for several useful discussions on this matter. Since there are 7 possible combinatorial types the assertion follows. $\square$

Our Halting Theorem is an immediate corollary of Lemmas 5.1 and 5.2. It says that we can stop looking for tiled discs if one hasn't shown up before a given time.

Halting Theorem 5.3 *Given an unknotted closed braid K of n strands and word length k in the Artin generators, let $t = 324n + 432k - 376$. Then there is an embedded disc in S^3 whose boundary is a closed braid conjugate to K and the induced braid foliation has complexity no higher than $(n,\ 28t2^{7t+2})$.*

Proof By the arguments which are reviewed in Section 2 of this paper, the normal surface can be isotoped to a tiled surface $\mathcal{D} = (D, G, C)$, via an isotopy which takes place outside a neighborhood of A. (If $\mathcal{D}$ is inessential, there is an essential tiled disc with fewer intersections with the braid axis.) As the isotopy is away from the braid axis, the points of intersection of the surface with A are unchanged.

By Lemma 5.1, the triangulation has at most $t = 324n+432k-376$ tetrahedra. By Lemma 5.2 the normal surface may be assumed to meet each of the $\leq t$ tetrahedron in at most $7t2^{7t+2}$ components. By Lemma 5.1 the braid axis meets 4 tetrahedra. By our construction of the triangulation in Section 5.1 the braid axis A passes through each of these as a line segment which meets each sheet of the normal surface at most once, transversally. The total number of intersections of A with $\mathcal{D}$ is then bounded above by $28t2^{7t+2}$. Thus the complexity (n, v) of the disc we seek is at most $(n,\ 28t2^{7t+2})$, where $t \leq 324n + 432k - 376$. $\square$

6 The algorithm

We summarize our test for whether K *is* the unknot in the form of an algorithm which is based on our work in earlier sections. We are given a knot K, which we may assume is given as a braid with n strands (see Appendix A) and Artin word length k. As in the previous section, let $t = 324n + 432k - 376$, so there is a triangulation of the knot complement with fewer than t tetrahedra.

Let $[K]$ denote the conjugacy class of the braid word given by K. The algorithm for determining whether K is the unknot is outlined below in Table 2 in pseudo-code.

```
(1) for each integer $P$ from $n$ to $14t2^{7t+2}+n$
(2)   for each positive embeddable tiled disc $\mathcal{D}'$
             with exactly $P$ positive vertices
(3)     for each way of adding $P-n$ negative vertices to $\mathcal{D}'$ to get $\mathcal{D}$
(4)       if $\mathcal{D}$ is embeddable and all b-arcs are essential
(5)         find the braid word $W(\partial\mathcal{D})$
(6)         compute its conjugacy class $[W(\partial\mathcal{D})]$
(7)         if $[W(\partial\mathcal{D})]=[K]$
(8)           $K$ is an unknot
(9)         endif
(10)      endif
(11)    next $\mathcal{D}$
(12)  next embeddable tiled disc
(13) next $P$
```

Table 2: The algorithm to test for unknottedness

Theorem 6.1 *The algorithm in Table 2 will enumerate a list of closed braids containing examples from each conjugacy class of closed braids representing the unknot. The given knot is unknotted if and only if it is in the conjugacy class of one of the braids on our list.*

Proof If K is unknotted, then the algorithm is guaranteed to find a disc that K bounds. Theorem 4.1 proves that lines (1), (2), (3) will generate all tiled discs which are embeddable tiled discs. Theorem 3.5 and Proposition 3.6 give the algorithmic tests for embeddability and essential b–arcs used in line (4). Theorem 3.4 explains how to read the braid word as needed in line (5). The solution to the conjugacy problem needed for lines (6) and (7) can be found in [4]. The Halting Theorem (Theorem 5.3) tells us that if K does bound a tiled disc, then that disc has no more than $28t2^{7t+2}$ vertices. Since it must have n more positive than negative vertices, we only need to check up to $14t2^{7t+2}+n$ positive vertices. □

Remark 6.2 The algorithm in Table 2 can easily be modified to generate examples. One could generate all unknotted braids with a certain size embeddable tiled disc, for example. In this way one could build some *unknot* tables which might speed up the work of others who are building *knot* tables. One could search for unknots whose embeddable tiled disc satisfies a particular property, such as not having any valence one vertices in the graph. Such examples are interesting in that they are the unknotted closed braids which have no trivial loop. One could search for any property of the embeddable tiled disc (including all properties of the braid word itself). We will discuss this more fully in our paper on the implementation of the algorithm.

Running Time and Implementation Issues This program has been implemented and is running, but not yet robustly tested. There are many aspects of an implementation that this paper has ignored, such as how to represent embeddable tiled discs on a computer, how to enumerate positive embeddable tiled discs, and how to decide where to add negative vertices. There are many solutions to these problems, and our current implementation is crude at best. When the entire program is in more mature form we will write a paper discussing these issues and giving experimental results.

Typically, one would like an estimate of running time for an algorithm. In this case it is best not contemplated. There are 2^{P-2} labelled trees (representing the positive embeddable tiled discs of the first step of the algorithm with cyclically ordered vertices) each of which has 2^{P-1} possible signs and $(P-2)!$ different cyclic orders on the singularities. Thus, the first part of the algorithm (generating all positive embeddable tiled discs of P vertices) has running time at of at least $P^{P-2}!(P-2)!2^{P-1}$, which assumes that everything is generated with perfect efficiency and no redundancy and doesn't take into account multiple, non-isomorphic embeddings of the graphs in the disc.

Adding the negative vertices is no faster, just harder to analyze. The good news is that there are many more non-embeddable or inessential ways to add negative vertices than embeddable, essential ways. Thus the number of embeddable tiled discs starts going down as negative vertices are added. The bad news is that the algorithm is still exponential time.

In practice, we don't bother with implementing the upper bound in line (1). Because of the exponential growth in the number of examples, this algorithm implemented on a single computer won't be able to go beyond, say, 20 vertices in any reasonable time.

The test for embeddability is quadratic in the number of vertices if implemented as written, as is the method for extracting the braid word. A faster version (in progress) of the embeddability test would be very helpful. The algorithm also depends heavily on the conjugacy computation of [4], and the running time of that computation has not yet been fully analyzed. The running time of the related solution to the word problem in [4] has been fully analysed and is shown in [4] to be $\mathcal{O}(|K|^2 n)$.

References

[1] **D Bennequin**, *Entrelacements et equations de Pfaff*, Asterisque 107–108 (1983) 87–161

[2] **J S Birman**, *Braids, Links and mapping class groups*, Annals of Math. Studies 82, Princeton University Press (1974)

[3] **J S Birman**, **E Finkelstein**, *Studying surfaces via closed braids*, International Journal of Knot Theory and its Applications, to appear

[4] **J S Birman**, **K H Ko**, **S J Lee**, *A new approach to the word and conjugacy problems in the braid groups*, Advances in Mathematics, to appear

[5] **J S Birman**, **W Menasco**, *Studying links via closed braids V: Closed braid representatives of the unlink*, Trans AMS, 329 (1992) 585–606

[6] **E A Elrifai**, **H Morton**, *Algorithms for positive braids*, Quart. J. Math. Oxford, 45 (1994) 479–497

[7] **F A Garside**, *The braid group and other groups*, Quart. J. Math. Oxford, 20 (1969) 235–254

[8] **W Haken**, *Theorie der Normalflachen*, Acta math. 105 (1961) 245–375

[9] **J Hass**, *Algorithms for recognizing knots and 3–manifolds*, Dept. of Math. UC Davis, preprint (1997)

[10] **J Hass**, **J Lagarias**, **N Pippenger**, *The computational complexity of knot and link problems*, preprint (1997)

[11] **F Jaeger**, **D L Vertigan**, **D J A Welsh**, *On the computational complexity of the Jones and Tutte polynomials*, Math. Proc. Camb. Phil. Soc. 108 (1990) 35–53

[12] **E S Kang**, **K H Ko**, **S J Lee**, *Band-generator presentation for the 4–braid group*, Topology and its Applications 78 (1997) 39–60

[13] **P Vogel**, *Representation of links by braids: A new algorithm*, Comment. Math Helv. 65 (1990) 104–113

[14] **P J Xu**, *The genus of closed 3–braids*, J. of Knot Theory and its Ramifications, 1 (1992) 303–326

[15] **S Yamada**, *The minimal number of Seifert circles equals the braid index of a link*, Invent. Math. 89 (1987) 347–356

A Appendix: Changing knots to closed braids

In spite of a flood of recent applications of braid theory to knot theory, many topologists regard the problem of changing knot diagrams to closed braids as a difficult project. We give, here, a very simple algorithm which does the job. It is due to Vogel [13], and incorporates earlier ideas of Yamada [15].

Let $D(K)$ be a c–crossing diagram on the plane $\mathbb{R}^2$ which describes a knot K. Smoothing the c crossings, the diagram is replaced by a *Seifert diagram*, ie by a collection of oriented Seifert circles $s_1, \ldots, s_n$ which are joined in pairs by half-twisted bands. We indicate the bands symbolically by signed *ties*, where the sign depends on the sign of the associated crossing. An example is given in Figure 19. (In the examples we omit the signs on the ties because they are irrelevant to the present discussion.)

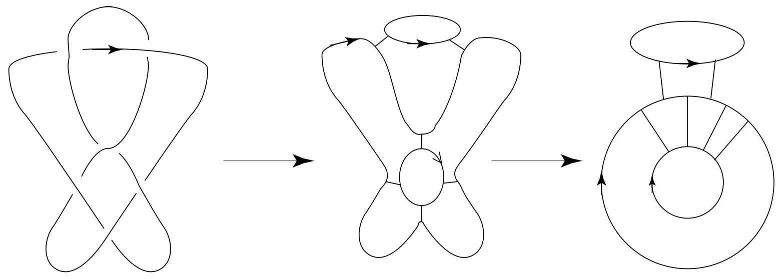


Figure 19: A Seifert diagram

Each pair of Seifert circles s_i, s_j cobounds a unique annulus $A_{i,j}$ on the 2–sphere $\mathbb{R}^2 \cup \{\infty\}$. Seifert circles s_i and s_j are *coherently oriented* if they represent the same element in $H_1(A_{i,j}; \mathbb{Z})$, otherwise they are incoherently oriented. A key fact, first noted by Yamada [15] is that if all pairs of Seifert circles are coherent, then the diagram is already essentially a closed braid diagram. This is the case in the example in Figure 19. Every pair of Seifert circles is coherently oriented, and there is no obstructions to sliding the ties together and grouping them into a single block. After so-doing we can number the strands and find a representing braid word. To be sure, it may be necessary to pull some of the Seifert circles through the point at infinity to obtain a more conventional closed braid diagram, as illustrated, but the braid word can be found without that final step, so the key point is to convert the given diagram to one in which every pair of Seifert circles is coherently oriented.

In the second example, given in Figure 20 there is an incoherent pair of Seifert circles.

Yamada proves that if an incoherent pair of Seifert circles exists, then one can always find a pair, such as the pair which is joined by the dotted arc in the second sketch for example (b). Call the circles s_p, s_q and call the arc $\alpha_{p,q}$. We require that the interior of $\alpha_{p,q}$ be disjoint from $D(K)$. Vogel's contribution was to notice that we can then use $\alpha_{p,q}$ to define a Reidemeister move of type II, as in the third sketch. This move

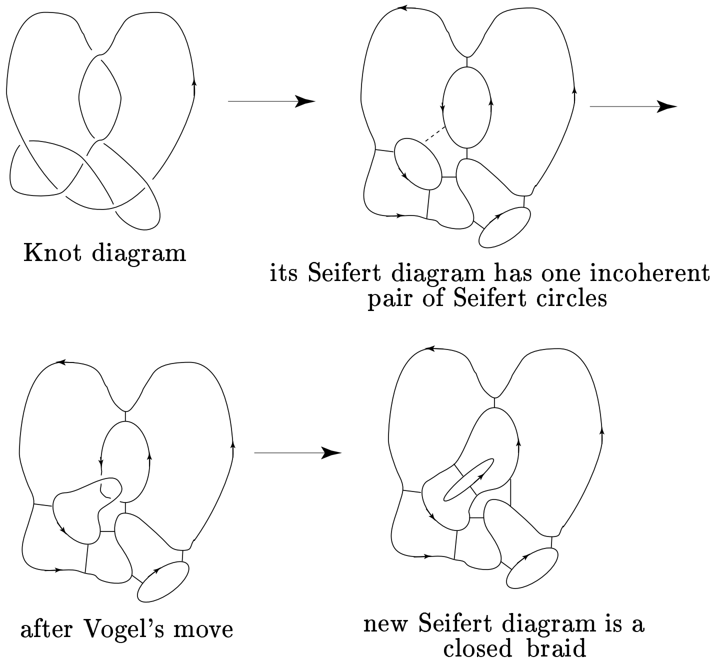


Figure 20: Another example

preserves the number of Seifert circles and reduces the number of incoherent pairs. Thus after a finite number of Vogel moves we will obtain a new diagram $D'(K)$ in which every pair of Seifert circles is coherently oriented.

Notice that the number n of Seifert circles is unchanged by the Vogel moves, and when all the Seifert circles are coherently oriented this number is the braid index of the given example. Hence, we may compute n from the given knot diagram of K. The writhe of the diagram is also unchanged by Vogel moves, and we finally obtain a closed braid the writhe is the number we have called the exponent sum. Thus, given a diagram of K we can easily convert it to a braid for use in our algorithm. The crossing number, is increased by Vogel moves, but it is proved in [13] that the increase is at most $(n-1)(n-2)$.

B Appendix: Testing for conjugacy

In this appendix we describe, briefly, the solution to the conjugacy problem which is given in [4]. We choose that solution over the other known solutions, for example [6], because it uses the band generators and so is natural for our work. Also, there is hope that when various technical difficulties are overcome it can be proved to be a polynomial-time algorithm.

Let $a_{t,s}$ be one of the band generators of the braid group which were introduced in Section 3 of this paper and illustrated in Figure 9. Notice that these generators include the more familiar generators $\{\sigma_1, \sigma_2, \ldots, \sigma_{n-1}\}$ as a proper subset, because $a_{i+1,i} = \sigma_i$. It is proved in [4] that:

Proposition B.1 *B_n has a presentation with generators $\{a_{ts};\ n \geq t > s \geq 1\}$ and with defining relations:*

(1) $a_{ts}a_{rq} = a_{rq}a_{ts}$ *if* $(t-r)(t-q)(s-r)(s-q) > 0$

(2) $a_{ts}a_{sr} = a_{tr}a_{ts} = a_{sr}a_{tr}$ *for all* t, s, r *with* $n \geq t > s > r \geq 1$.

The defining relations in the presentation of Theorem B.1 for B_n, like the more standard presentation of Artin, only involves positive powers of the generators. This allows us to introduce a semigroup B_n^+ which has the same presentation as B_n in terms of the band generators. There is a natural map $B_n^+ \to B_n$ which takes each generator $a_{t,s}$ of B_n^+ to the corresponding generator of B_n. Following methods pioneered by Garside, the following embedding theorem is proved in [4]. We remark that the theorem does not appear to follow directly from the earlier work of Garside.

Proposition B.2 *If two positive words in the band generators represent the same element of B_n, then they also represent the same element of B_n^+.*

In view of Proposition B.2, we may regard B_n^+ as a submonoid of B_n. Generalizing the ideas of Garside (but many of the details are different because of the new generating set) the *fundamental word* δ is introduced in [4]:

$$\delta = a_{n(n-1)}a_{(n-1)(n-2)}\cdots a_{21}. \tag{1}$$

The reader who is familiar with the mathematics of braids will recognize that δ^n generates the center of B_n. The following two properties of δ are established in [4].

Lemma B.3 *Let δ be the fundamental braid. Then:*

(1) *For each generator $a_{t,s}$ of B_n there exists a positive word $P_{t,s}$ in the band generators such that $a_{t,s}^{-1} = \delta^{-1}P_{t,s}$.*

(2) *The braid δ has the following (weak) commutativity properties: $a_{t,s}\delta = \delta a_{t+1,s+1}$ and $a_{t,s}\delta^{-1} = \delta^{-1}a_{t-1,s-1}$*

Using (1) of Lemma B.3, we may replace all negative letters in any word W in the band generators by positive words and powers of δ^{-1}. Using (2) we may then move all powers of δ^{-1} to the left. In this way, one may choose an arbitrary word W in the band generators and change it to an equivalent word of the form $\delta^p P$, where $P \in B_n^+$. An element of B_n^+ is a *descending cycle* if it can be represented by a word of the form

$$a_{(t_k,t_{k-1})}a_{(t_{k-1},t_{k-2})},\ldots,a_{(t_2,t_1)}, \quad n \geq t_k > t_{k-1} > \ldots > t_1 \geq 1.$$

We describe a descending cycle by the array of subscripts $(t_k, t_{k-1}, \ldots, t_1)$. Two descending cycles $(t_k, t_{k-1}, \ldots, t_1)$ and $(s_m, s_{m-1}, \ldots, s_1)$ are *parallel* if no pair (t_j, t_i) separates any pair (s_q, s_p). These concepts are investigated in detail in [4]. A *canonical factor* is an element of B_n^+ which can be represented by a product of parallel descending cycles. It is proved in [4] that the canonical factors are precisely the 'initial segments' of δ, a concept which will be familiar to those readers who have worked through the details of the papers on Garside's algorithm. The main result in [4] is:

Theorem B.4 [4] *Let W be an arbitrary word in the band generators which represents $\mathcal{W} \in B_n$. Then there is a algorithmic procedure which allows one to find, starting with W, all possible representatives of the conjugacy class $[\mathcal{W}]$ of the form $\delta^p A_1 A_2 \ldots A_k$, such that:*

(1) *p is maximal for all such representations,*

and simultaneously

(2) *k is minimal for all such representations.*

Also:

(3) *each $A_i \in B_n^+$ is a canonical factor,*

(4) *each product $A_i A_{i+1}$, $i = 1, \ldots, k-1$ is left-weighted, as defined in [4].*

The set of finitely many words in the form $\delta^p A_1 A_2 \ldots A_k$ which represent $\mathcal{W}$ is known as the super summit set of $[\mathcal{W}]$. Two elements $\mathcal{W}, \mathcal{W}' \in B_n$ are conjugate if and only if they have the same values of p and k and the same super summit set.