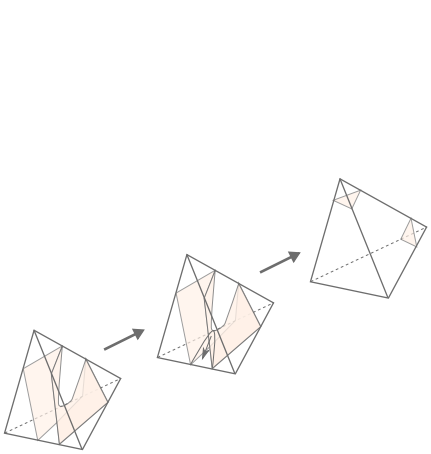
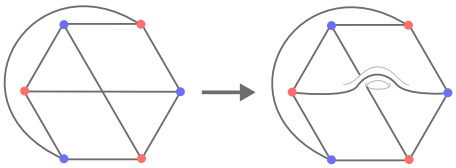
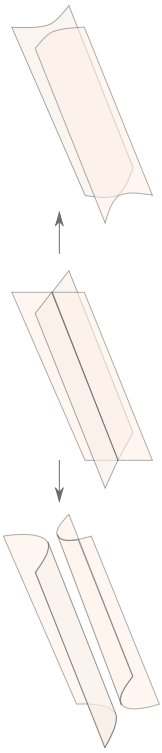
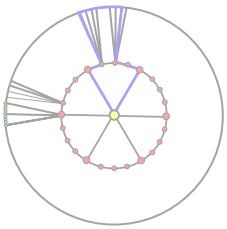
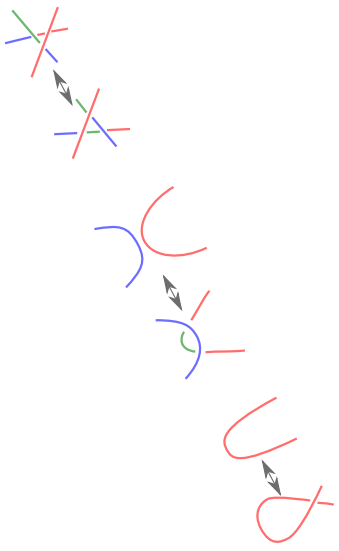
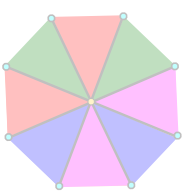
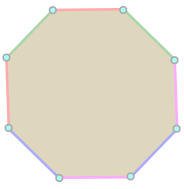
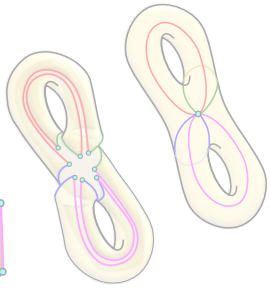




Computational Topology
Lecture notes

Francis Lazarus and Arnaud de Mesmay

2016-2018

Contents

1

Panorama

Contents

1.0.1 Topology.

Topology deals with the study of *spaces*. One of its goals is to answer the following broad class of questions:

"Are these two spaces the same?"

This naturally leads to the following subquestions:

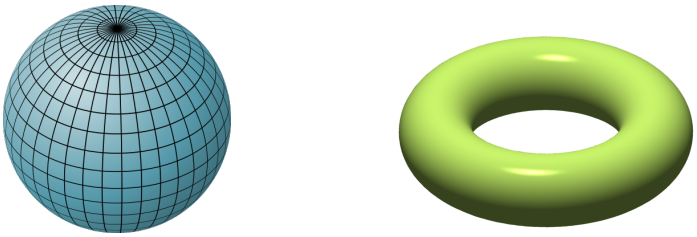
- **What is a space?** General topology typically defines topological spaces via open and closed sets. In order to avoid pathological examples, and with an eye towards applications, we will take a more concrete approach[1]: in this course, topological spaces will be obtained in the form of **complexes**, that is, by gluing together fundamental blocks. For example, gluing segments yields a graph, while by gluing together triangles one can obtain a surface (or something more complicated). The usual notions of distance on these fundamental blocks naturally induce a notion of proximity on such a complex, and therefore a topology whose properties are convenient to understand geometrically.

[1] This is by no means original: see introductory textbooks on algebraic topology, for example Hatcher [Hat02] or Stillwell [Sti93].

- **What is "the same" ?** It very much depends on the context. The most common equivalence relation is **homeomorphism**, which is a continuous map with a continuous inverse function. But in some contexts, when a space is **embedded** in another space, one will be interested in distinguishing between different embeddings. There, a convenient notion is **isotopy** : two embedded spaces will be considered the same if one can deform continuously one into the other one.

Let us look at examples.

Example 1: By gluing triangles or quadrilaterals, one can easily obtain a sphere (left figure), or a torus (right figure).

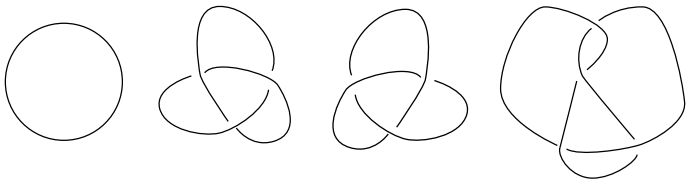


Are these two spaces homeomorphic? Obviously not: the torus has a hole. But what is a hole? Two naive answers will guide us to the two fundamental constructs of algebraic topology:

- **Homotopy**: On the sphere, every closed curve can be deformed into a single point. While on the torus, a curve going around the hole can not. Such a curve is not **homotopic** to a point.
- **Homology**: On the sphere, every closed curve separates the sphere into two regions. While on the torus, a curve going around the hole is not separating. Such a curve is not trivial in **homology**.

These intuitions can be formalized into algebraic objects which will constitute invariants (actually, **functors**) that one can use to distinguish topological spaces.

Example 2: By gluing segments in $\mathbb{R}^3$, one can obtain the following **knots**.



Are they homeomorphic? Certainly: they are both homeomorphic to the circle $\mathbb{S}^1$. But are they **isotopic**: can one be deformed into the other without crossing itself? The answer is negative, but this is not that easy to prove. One way to see it is that the knot on the left bounds a disk, while the one on the right does not. Studying which surfaces one can find in a 3-dimensional space is the goal of **normal surface theory**.

1.0.2 Computational Topology.

Computational topology deals with effective computations on topological spaces. The main question now becomes:

"How to compute whether these two spaces are the same?"

Note that since we study spaces described by gluings of fundamental blocks, in most instances this can be made into a well-defined algorithmic problem, with a finite input. One can then wonder about the **complexity** of this problem, and aim to design the most efficient algorithm, or conversely prove hardness results. Throughout the course, we will investigate the complexity of various instances of this question, with practical algorithms computing homeomorphism, homotopy, homology or isotopy for example.

Outline. We will work by increasing progressively the dimension, and thus the complexity of the objects we consider.

1. We start with one of the simplest topological spaces : the **plane** $\mathbb{R}^2$. Describing it as a union of small blocks amounts to the study of **planar graphs**. This topological constraint on graphs has a strong impact on their combinatorics, which we will study through various angles.

2. Next come **surfaces**, which look locally like the plane. From a mathematical point of view, these are still fairly simple, as they can be easily classified. But once again, there is a very fruitful interplay between the topology of surfaces and the combinatorics of embedded graphs. Moreover, surfaces are a convenient and easy framework to introduce **homotopy** and **homology** and we will present efficient algorithms for the computation of these invariants.

3. In dimension 3, we will introduce **knots** and **3-manifolds**. Distinguishing various knots is hard: the whole field of **knot theory** is dedicated to this. We will see through various examples why this is hard, and will introduce **normal surface theory**, one of the main tools used for computational problems in 3 dimensions. As an application, we will use it to provide an algorithm to recognize trivial knots in **NP**.

4. As soon as we hit dimension 4, we start to hit the **limits** of computational topology: many problems are not only hard, they are **undecidable**. We will introduce **simplicial complexes**, which are the main model for high-dimensional topological spaces, and show that deciding homeomorphism of such complexes is already undecidable in dimension 4, as is testing the homotopy of curves in 2-dimensional complexes.

5. Although computing homotopy and homeomorphism quickly becomes intractable in high dimensions, homology does not: its simple algebraic structure allows

for efficient computations that scale well with the dimension. This can be leveraged as a tool for big data: the techniques of **topological data analysis** aim at recognizing topological features in point clouds by computing their **persistent homology**. As we shall see, this is a surprisingly powerful way to infer information from a structured point cloud.

Applications. The approach in this course is to focus on the mathematical motivations to study topological objects and their computation. This does not mean that this is all devoid of applications. Quite the contrary: topological spaces are ubiquitous in computer science, and the primitives we develop here have practical implications in computer graphics [LGQ09], mesh processing [GW01], robotics [Far08], combinatorial optimization (see references in [Eri12]), machine learning [ACC16], and many other fields. Believe it or not, they even revolutionized basketball [Bec12]!

2

Planar Graphs

Contents

A graph is planar if it can be drawn on a sheet of paper so that no two edges intersect, except at common endpoints. This simple property not only allows to visualize planar graphs easily, but implies many nice properties. Planar graphs are sparse: they have a linear number of edges with respect to their number of vertices (specifically a simple planar graph with n vertices has at most $3n-6$ edges), they are 4-colorable, they can be encoded efficiently, etc. Classical examples of planar graphs include the graphs formed by the vertices and edges of the five Platonic polyhedra, and in fact of any convex polyhedron. Although being planar is a topological property, planar graphs have purely combinatorial characterizations. Such characterizations may lead to efficient algorithms for planarity testing or, more surprisingly, for geometric embedding (=drawing).

In the first part of this lecture we shall deduce the combinatorial characterizations of planar graphs from their topological definition. That we can get rid of topological considerations should not be surprising. It is actually possible to develop a combinatorial theory of surfaces where a drawing of a graph is *defined* by a circular ordering of its edges around each vertex. The collection of these circular orderings is called a **rotation**

system. A rotation system is thus described combinatorially by a single permutation over the (half-)edges of a graph; the cycle decomposition of the permutation induces the circular orderings around each vertex. The topology of the surface corresponding to a rotation system can be deduced from the computation of its Euler characteristic. Being planar then reduces to the existence of a rotation system with the appropriate Euler characteristic.

The following notes are largely inspired by the monographs of Mohar and Thomassen [MT01] and of Diestel [Die05].

2.1 Topology

A **graph** $G=(V,E)$ is defined by a set $V=V(G)$ of vertices and a set $E=E(G)$ of edges where each edge is associated one or two vertices, called its **endpoints**. A **loop** is an edge with a single endpoint. Edges sharing the same endpoints are said **parallel** and define a **multiple edge**. A graph without loops or multiple edges is said **simple** or **simplicial**. In a simple graph every edge is identified unambiguously with the pair of its endpoints. Edges should be formally considered as pairs of oppositely oriented **arcs**. A **path** is an alternating sequence of vertices and arcs such that every arc is preceded by its origin vertex and followed by the origin of its opposite arc. A path may have repeated vertices (beware that this is not standard, and usually called a **walk** in graph theory books). Two or more paths are **independent** if none contains an inner vertex of another. A **circuit** is a closed path, *i.e.* a path whose first and last vertex coincide. A **cycle** is a simple circuit (without repeated vertices). We will restrict to finite graphs for which V and E are finite sets.

The Euclidean distance in the plane $\mathbb{R}^2$ induces the usual topology where a subset $X\subset\mathbb{R}^2$ is **open** if every of its points is contained in a ball that is itself included in X. The **closure** $\bar{X}$ of X is the set of limit points of sequences of points of X. The **interior** $\mathring{X}$ of X is the union of the open balls contained in X. An **embedding** of a non-loop edge in the plane is just a topological embedding (a homeomorphism onto its image) of the segment $[0,1]$ into $\mathbb{R}^2$. Likewise, an embedding of a loop-edge is an embedding of the circle $S^1=\mathbb{R}/\mathbb{Z}$. An **embedding** of a finite graph $G=(V,E)$ in the plane is defined by a 1-1 map $V\hookrightarrow\mathbb{R}^2$ and, for each edge $e\in E$, by an embedding of e sending $\{0,1\}$ to e's endpoints such that the relative interior of e (the image of $]0,1[$) is disjoint from other edge embeddings and vertices[1].

A graph is **planar** if it has an embedding into the plane. Thanks to the stereographic projection, the plane can be equivalently replaced by the sphere. A **plane graph** is a specific embedding of a planar graph. A connected plane graph in the plane has a single unbounded face. In contrast, all the faces play the same role in an embedding into the sphere and any face can be sent to the unbounded face of a plane embedding by a stereographic projection.

As far as planarity is concerned we can restrict to simple graphs. Indeed, it is easily seen that a graph has an embedding in the plane if and only if this is the case for the graph obtained by removing loop edges and replacing each multiple edge by a single

[1] In other words, this is a topological embedding of the quotient space $(V\sqcup[0,1]\times E)/\sim$, where $\sim$ identifies edge extremities $(0,e)$ and $(1,e)$ with the corresponding vertices.

edge. When each edge embedding $[0,1] \to \mathbb{R}^2$ is piecewise linear the embedding is said **PL**, or **polygonal**. A **straight line embedding** corresponds to the case where each edge is a line segment.

Lemma 2.1.1. *A graph is planar if and only if it admits a PL embedding.*

The proof is left as an exercise. One can first show that a connected subset of the plane is connected by simple PL arcs.

2.1.1 The Jordan Curve Theorem

Most of the facts about planar graphs ultimately relies on the Jordan curve theorem, one of the most emblematic results in topology. Its statement is intuitively obvious: a simple closed curves cuts the plane into two connected parts. Its proof is nonetheless far from obvious, unless one appeals to more advanced arguments of algebraic topology. Camille Jordan (1838 – 1922) himself proposed a proof whose validity is still subject of debates [Hal07b]. A rather accessible proof was proposed by Helge Tverberg [Tve80] (see the course notes [Laz12] for a gentle introduction). Eventually, a formal proof was given by Thomas Hales (and other mathematicians) [Hal07b, Hal07a] and was automatically checked by a computer. Concerning the Jordan-Schoenflies theorem, the situation is even worse. This stronger version of the Jordan curve theorem asserts that a simple curve does not only cut a sphere into two pieces but that each piece is actually a topological disc. A nice proof by elementary means – but far from simple – and resorting to the fact that $K_{3,3}$ is not planar is due to Carsten Thomassen [Tho92].

The main source of difficulties in the proof of the Jordan curve theorem is that a continuous curve can be quite wild, e.g. fractal. When dealing with PL curves only, the theorem becomes much easier to prove.

Theorem 2.1.2 (Polygonal Jordan curve —)**.** *Let C be a simple closed PL curve. Its complement $\mathbb{R}^2 \setminus C$ has two connected components, one of which is bounded and each of which has C as boundary.*

PROOF. Since C is contained in a compact ball, its complement has exactly one unbounded component. Define the *horizontal rightward* direction $\vec{h}$ as some fixed direction transverse to the all the line segments of C. For every segment s of C we let $\underline{s}$ be the lower half-open segment obtained from s by removing its upper endpoint. We also denote by h_p the ray with direction $\vec{h}$ starting at a point $p \in \mathbb{R}^2$. We consider the *parity function* $\pi : \mathbb{R}^2 \setminus C \to \{$ even, odd $\}$ that counts the parity of the number of lower half-open segments of C intersected by a ray:

$$\pi(p) := \text{ parity of } \left| \{ \text{ segment } s \text{ of } C \mid h_p \cap \underline{s} \neq \emptyset \} \right|$$

Every $p \in \mathbb{R}^2 \setminus C$ is the center of small disk D_p over which π is constant. Indeed, let S_p be the set of segments (of C) that avoid h_p, let S'_p be the set of segments whose interior crosses h_p and let S''_p be the set of segments whose lower endpoint lies on h_p. If D_p

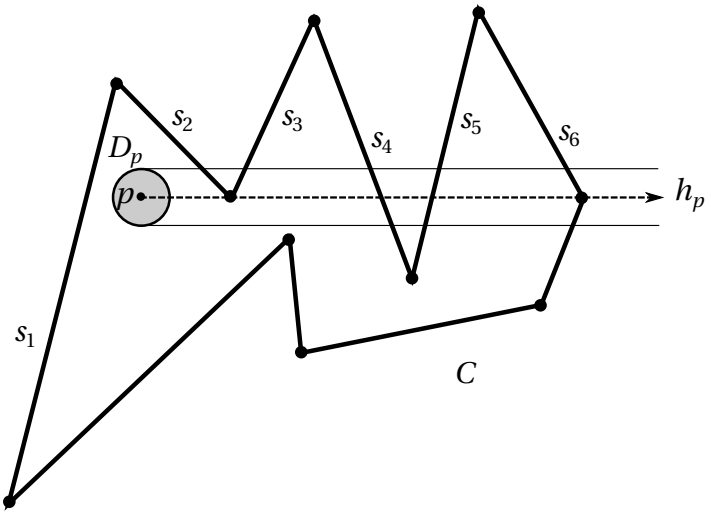


Figure 2.1: The horizontal ray through p cuts the five lower half-open segments $\underline{s}_2, \underline{s}_3, \underline{s}_4, \underline{s}_5, \underline{s}_6$. Here, we have $s_1 \in S_p$, $s_4, s_5 \in S'_p$ and $s_2, s_3, s_6 \in S''_p$.

is sufficiently small, then for every $q \in D_p$ we have $S_q = S_p$, $S'_q = S'_p$ and the parity of $|S''_q|$ and $|S''_p|$ is the same. See Figure 2.1. It follows that $\pi(q) = \pi(p)$. Since π is locally constant, it must be constant over each connected component of $\mathbb{R}^2 \setminus C$. Moreover, the parity function must take distinct values on points close to C that lie on a same horizontal but on each side of C. It follows that $\mathbb{R}^2 \setminus C$ has at least two components. To see that $\mathbb{R}^2 \setminus C$ has at most two components consider a small disk D centered at a point interior to a segment s of C. Then $D \setminus C = D \setminus s$ has two components. Moreover, any point in $\mathbb{R}^2 \setminus C$ can be joined to one of these components by a polygonal path that avoids C: first come close to C with a straight line and then follow C in parallel until D is reached. Finally, it is easily seen by similar arguments as above that every point of C is in the closure of both components of $\mathbb{R}^2 \setminus C$. □

Corollary 2.1.3 (θ's lemma)**.** *Let C_1, C_2, C_3 be three simple PL paths with the same endpoints p, q and otherwise disjoint. The graph $G = C_1 \cup C_2 \cup C_3$ has three faces bounded by $C_1 \cup C_2$, $C_2 \cup C_3$ and $C_3 \cup C_1$, respectively.*

PROOF. From the Jordan curve theorem the three simple closed curves $G_k = C_i \cup C_j$, $\{i, j, k\} = \{1, 2, 3\}$, cut the plane into two components bounded by G_k. We let X_k and Y_k be respectively the bounded and unbounded component. We also denote by $\overset{\circ}{C}_i := C_i \setminus \{p, q\}$ the relative interior of C_i. We first remark that a simple PL path cuts an open connected subset of the plane into at most two components: as in the proof of the Jordan curve theorem we can first come close to the path and follow it until a small fixed disk is reached. Since $\overset{\circ}{C}_3$ is included in a face of G_3, we deduce that $G = G_3 \cup \overset{\circ}{C}_3$ has at most three faces.

We claim that $\overset{\circ}{C}_i \subset X_i$ for at least one index $i \in \{1, 2, 3\}$. Otherwise we would have $C_i \subset \complement X_i$, whence $G \subset \complement X_i$, or equivalently $X_i \subset \complement G$. So, X_i would be a face of G. Since the X_i's are pairwise distinct (note that $C_i \subset \bar{X}_j$ while $\overset{\circ}{C}_i \not\subset \bar{X}_i$), we would infer that G

has at least three bounded faces, hence at least four faces. This would contradict the first part of the proof. Without loss of generality we now assume $\overset{\circ}{C}_3 \subset X_3$.

From $G = G_1 \cup G_2$ we get that each face of G is a component of the intersection of a face of G_1 with a face of G_2. From $G_3 \subset G \subset \complement Y_3$ we get that Y_3 is a face of G. Since Y_3 is unbounded we must have $Y_3 \subset Y_1 \cap Y_2$.

Now, $C_1 \subset \bar{Y}_3 \subset \bar{Y}_1 = \complement X_1$ implies $G = G_1 \cup C_1 \subset \complement X_1$. It follows that X_1 is a face of G. Likewise, X_2 is a face of G. Moreover, these two faces are distinct (C_1 bounds X_1 but not X_2). We conclude that Y_3, X_1 and X_2 are the three faces of G. □

2.1.2 Euler's Formula

The famous formula relating the number of vertices, edges and faces of a plane graph is credited to Leonhard Euler (1707-1783) although René Descartes had already deduced very close relations for the graph of a convex polyhedron. See the historical account of R. J. Wilson in [Jam99, Sec. 17.3] and in J. Erickson's course notes http://jeffe.cs.illinois.edu/teaching/topology17/chapters/02-planar-graphs.pdf

Recall that a graph G is **2-connected** if it contains at least three vertices and if removing any one of its vertices leaves a connected graph. If G is 2-connected, it can be constructed by iteratively adding paths to a cycle. In other words, there must be a sequence of graphs $G_0, G_1, \dots, G_k = G$ such that G_0 is a cycle and G_i is deduced from G_{i-1} by attaching a simple path between two distinct vertices of G_{i-1}.

Proposition 2.1.4. *Each face of a 2-connected PL plane graph is bounded by a cycle of the graph. Moreover, each edge is incident to (= is in the closure of) exactly two faces.*

PROOF. Let G be a 2-connected PL plane graph. Consider the sequence $G_0, G_1, \dots, G_k = G$ as above. We prove the proposition by induction on k. If $k = 0$, then G is a cycle and the proposition reduces to the Jordan curve theorem 2.1.2. Otherwise, by the induction hypothesis G_{k-1} satisfies the proposition. Let P be the attached path such that $G = G_{k-1} \cup P$. The relative interior of P must be contained in a face f of G_{k-1}. This face is bounded by a cycle C of G_{k-1}. We can now apply θ's lemma 2.1.3 to $C \cup P$ and conclude that f is cut by G into two faces bounded by the cycles $C_1 \cup P$ and $C_2 \cup P$, where C_1, C_2 are the subpaths of C cut by the endpoints of P. Moreover all the other faces of G_{k-1} are faces of G bounded by the same cycles. It easily follows that the edges of G are each incident to exactly two faces. □

Lemma 2.1.5. *Let G be a PL plane graph. If v is a vertex of degree one in G then $G - v$ and G have the same number of faces.*

PROOF. We denote by e the edge incident to v in G. Every face of G is contained in a face of $G - v$. Moreover, the relative interior of (the embedding of) e is contained in a face f of $G - v$. Hence, every other face of $G - v$ is also a face of G. It remains to count the number of faces of G in f. Let p, p' be two points in $f \setminus e$. There is a PL path in f connecting p and p'. This path may intersect e, but we may avoid this intersection by considering a detour in a small neighborhood N_e of e in f (indeed,

$N_e \setminus e$ is connected). It follows that p and p' belong to a same component of $f \setminus e$. We conclude that G has only one face in f, so that G and $G-v$ have the same number of faces. □

Theorem 2.1.6 (Euler's formula)**.** *Let $|V|,|E|$ and $|F|$ be the number of vertices, edges and faces of a connected plane graph G. Then,*

$$|V|-|E|+|F|=2$$

PROOF. We argue by induction on $|E|$. If G has no edges then it has a single vertex and the above formula is trivial. Otherwise, suppose that G has a vertex v of degree one. Then by Lemma 2.1.5, G has the same number of faces as $G-v$. Note that G has one vertex more and one edge more than $G-v$. By the induction hypothesis we can apply Euler's formula to $G-v$, from which we immediately infer the validity of Euler's formula for G. If every vertex of G has degree at least two, then G contains a cycle C. Let e be an edge of C. We claim that G has one face more than $G-e$. This will allow to conclude the theorem by applying Euler's formula to $G-e$, noting that G has the same number of vertices but one edge less than $G-e$. By the Jordan curve theorem 2.1.2, C cuts the plane into two faces (components) bounded by C. Since $G=C\cup(G-e)$, every face of G is included in the intersection of a face of C and a face of $G-e$. Let f be the face of $G-e$ containing the relative interior of e. Every other face of $G-e$ does not meet C, hence is also a face of G. Since f intersects the two faces of C (both bounded by e), G has at least one face more than $G-e$. By considering a small tubular neighborhood of e in f, one shows by an already seen argument that $f \setminus e$ has at most two components. It follows that f contains exactly two faces of G, which concludes the claim. □

Application. Two old puzzles that go back at least to the nineteenth century are related to planarity and can be solved using Euler's formula. The first asks whether it is possible to divide a kingdom into five regions so that each region shares a frontier line with each of the four other regions. The second puzzle, sometimes called the *gaz-water-electricity problem* requires to join three houses to three gaz, water and electricity facilities using pipes so that no two pipes cross. By duality, the first puzzle translates to the question of the planarity of the **complete graph** K_5 obtained by connecting five vertices in all possible ways. The second problem reduces to the planarity of the **complete bipartite graph** $K_{3,3}$ obtained by connecting each of three independent vertices to each of three other independent vertices. It appears that these two puzzles are unfeasible.

Theorem 2.1.7. *K_5 and $K_{3,3}$ are not planar.*

PROOF. We give two proofs. The first one is based on Euler's formula.

1. Suppose by way of contradiction that $K_{3,3}$ has a plane embedding. Euler's formula directly implies that the embedding has $n=2-6+9=5$ faces. Since $K_{3,3}$ is 2-connected, it follows from Proposition 2.1.4 that every edge is incident to two

distinct faces. By the same proposition, each face is bounded by a cycle, hence by at least 4 edges (cycles in a bipartite graph have even lengths). It follows from the *handshaking lemma* that twice the number of edges is larger than four times the number of faces, *i.e.* $18 \geq 20$. A contradiction.

An analogous argument for K_5 implies that an embedding must have 7 faces. Since every face is incident to at least 3 edges, we infer that $2 \times 10 \geq 3 \times 7$. Another contradiction.

2. Let $\{1,3,5\}$ and $\{2,4,6\}$ be the two vertex parts of $K_{3,3}$. The cycle $(1,2,3,4,5,6)$ separates the plane into two components in any plane embedding of $K_{3,3}$. By θ's lemma the edges $(1,4)$ and $(2,5)$ must lie in the face that does not contain $(3,6)$. Then $(1,4)$ and $(2,5)$ intersect, a contradiction. A similar argument applies for the non-planarity of K_5.

□

Exercise 2.1.8. Every simple planar graph G with $n \geq 3$ vertices has at most $3n-6$ edges and at most $2n-4$ faces.

Exercise 2.1.9. Every simple planar graph with at least six vertices has a vertex with degree less than 6.

To conclude, we prove a very strong generalization of Exercise 2.1.8, which allows to quantify how non-planar dense graphs are. Here, a **drawing** of a graph is just a continuous map $f : G \to \mathbb{R}^2$, that is, a drawing of the graph on the plane where crossings are allowed. The **crossing number** $cr(G)$ of a graph is the minimal number of crossings over all possible drawings of G. For instance, $cr(G) = 0$ if and only if G is planar. The **crossing number inequality** [ACNS82, Lei84] provides the following lower bound on the crossing number.

Theorem 2.1.10. $cr(G) \geq \frac{|E|^3}{64|V|^2}$ *if* $|E| \geq 4|V|$.

The proof is a surprising application of (basic) probabilistic tools.

PROOF. Starting with a drawing of G with the minimal number of crossings, define a new graph G' obtained by removing one edge for each crossing. This graph is planar since we removed all the crossings, and it has at least $|E| - cr(G)$ edges (removing one edge may remove more than one crossing), so we obtain that $|E| - cr(G) \leq 3|V|$. (Note that we removed the -6 to obtain an inequality valid for any number of vertices.) This gives in turn

$$cr(G) \geq |E| - 3|V|.$$

This can be amplified in the following way. Starting from G, define another graph by removing vertices (and the edges adjacent to them) at random with some probability $1-p < 1$, and denote by G'' the obtained graph. Taking the previous inequality with expectations, we obtain $\mathbb{E}(cr(G'')) \geq \mathbb{E}(|E''|) - 3\mathbb{E}(|V''|)$. Since vertices are removed with probability $1-p$, we have $\mathbb{E}(|V''|) = p|V|$. An edge survives if and only if both its endpoints survive, and a crossing survives if and only if the four adjacent vertices survive (there may be less than four adjacent vertices in general, but not in the drawing

minimizing the crossing number, we leave this as an exercise to check), so we get $\mathbb{E}(|E''|) = p^2|E|$ and $\mathbb{E}(cr(G'')) = p^4 cr(G)$. So we obtain

$$cr(G) \geq p^{-2}|E| - 3p^{-3}|V|,$$

and taking $p = 4|V|/|E|$ – which is less than 1 if $|E| \geq 4|V|$ – gives the result. □

2.2 Kuratowski's Theorem

2.2.1 The Subdivision Version

We say that H is **subdivision** of G if H is obtained by replacing the edges of G by independent simple paths of one or more edges. Obviously, a subdivision of a non-planar graph is also non-planar. It follows from Theorem 2.1.7 that a planar graph cannot have a subdivision of K_5 or $K_{3,3}$ as a subgraph. In 1929, Kazimierz Kuratowski (1896 – 1980) succeeded to prove that this condition is actually sufficient for a graph to be planar. For this reason K_5 and $K_{3,3}$ are called the *Kuratowski graphs*, or the *forbidden graphs*.

Theorem 2.2.1 (Kuratowski, 1929)**.** *A graph is planar if and only if it does not contain a subdivision of K_5 or $K_{3,3}$ as a subgraph.*

As just noted, we only need to show that a graph without any subdivision of a forbidden graph is planar. We follow the proof of Thomassen [MT01]. Recall that a graph is **3-connected** if it contains at least four vertices and if removing any two of its vertices leaves a connected graph. By Menger's theorem [Wil96, cor. 28.4], a graph is 3-connected if and only if any two distinct vertices can be connected by at least three independent paths. If e is an edge of a graph G we denote by $G/\!/e$ the graph obtained by the **contraction** of e, *i.e.* by deleting e, identifying its endpoints, and merging each resulting multiple edge, if any, into a single edge. The proof of Kuratowski's theorem first restricts to 3-connected graphs. By Lemma 2.2.2 below we can repeatedly contract edges while maintaining the 3-connectivity until the graph is small enough so that it can be trivially embedded into the plane. We then undo the edge contractions one by one and construct corresponding embeddings. In the end, the existence of an embedding attests the planarity of the graph. In a second phase we extend the theorem to any graph, not necessarily 3-connected, that does not contain any subdivision of K_5 or $K_{3,3}$. This is done by adding as many edges as possible to the graph without introducing a (subdivision of a) forbidden graph. By Lemma 2.2.5 below the resulting graph is 3-connected and we may conclude with the first part of the proof.

Lemma 2.2.2. *Any 3-connected graph G with at least five vertices contains an edge e such that $G/\!/e$ is 3-connected.*

PROOF. Suppose for the sake of contradiction that for any edge $e = xy$, the graph $G/\!/e$ is not 3-connected. Denote by v_e the vertex of $G/\!/e$ resulting from the identification of x and y. Then we can find a vertex $z \in V(G/\!/e)$ such that $\{z, v_e\}$ disconnects

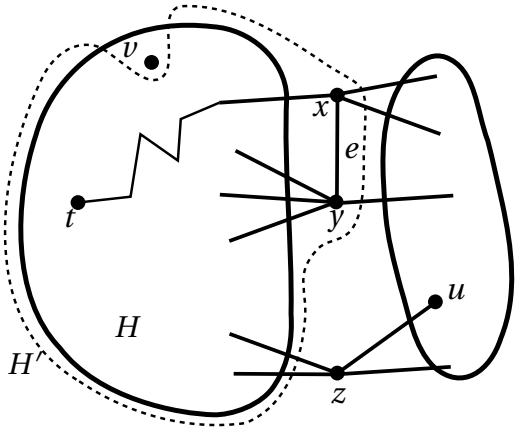


Figure 2.2: H' has more vertices than H even when $v \in \{x, y\}$.

$G /\!/ e$. In other words, for any edge $e = xy$ of G we can find a vertex $z \in V(G)$ such that $G - \{x, y, z\}$ is not connected. We choose e and z such that the number of vertices of the largest component, say H, of $G - \{x, y, z\}$ is maximal. Let u be adjacent to z in a component of $G - \{x, y, z\}$ other than H. See figure 2.2. By the above reformulation, we can find a vertex $v \in V(G)$ such that $G - \{z, u, v\}$ is not connected. We claim that the subgraph H' induced by $(V(H) \cup \{x, y\}) \setminus \{v\}$ is connected. Since H' is contained in $G - \{z, u, v\}$ and since H' has more vertices than H, this contradicts the choice of H, hence concludes the proof. To see that H' is connected we just need to check that every $t \in V(H)$ can be connected to x or y (themselves connected by e) by a path in H'. Since G is 3-connected, there is a path $p : t \rightsquigarrow x$ in G avoiding z and v. Replacing x by y if necessary, we can assume that $p - x$ does not contain y. It follows that $p - x$ is contained in $G - \{x, y, z, v\}$, hence in $H - v$. So p is in H'. □

Exercise 2.2.3. Let e be an edge of G such that $G /\!/ e$ contains a subdivision of a forbidden graph. Show that G already contains such a subdivision. (Hint: $G /\!/ e$ and G need not contain subdivisions of the same forbidden graph.)

A straight line embedding is said **convex** if all its faces are bounded by convex polygons.

Proposition 2.2.4 (Kuratowski's theorem for 3-connected graphs)**.** *A 3-connected graph G without any subgraph isomorphic to a subdivision of a forbidden graph admits a convex embedding.*

PROOF. We use induction on the number of vertices of G. The proposition is easily checked by hand if G has four vertices. Otherwise, G has at least five vertices, and by Lemma 2.2.2 we may choose an edge $e = xy$ such that $G' := G /\!/ e$ is 3-connected. Moreover, G' contains no subdivision of a forbidden graph. See Exercise 2.2.3. By induction, G' has a convex embedding. Let z be the vertex of G' resulting from the identification of x and y. Since $G' - z$ is 2-connected, we know by Proposition 2.1.4 that the face of $G' - z$ that contains z is bounded by a cycle C of G. Let $X = \{u \in V(G) \mid ux \in E(G)\}$ and let $Y = \{u \in V(G) \mid uy \in E(G)\}$. We claim that X and Y are *not interleaved* in C, *i.e.* $|X \cap Y| \leq 2$ and we cannot find two vertices in X and two vertices

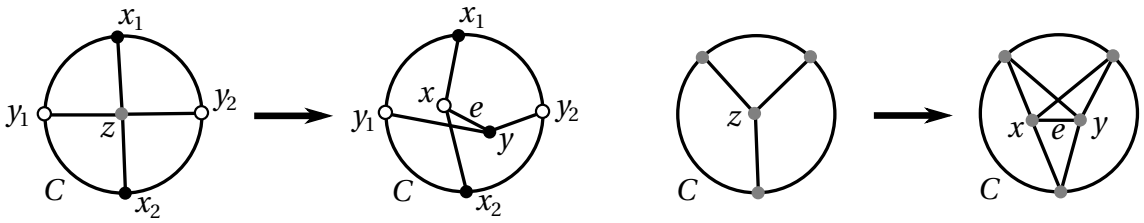


Figure 2.3: Left, two vertices $x_1, x_2 \in X$ and two vertices $y_1, y_2 \in Y$ appear in an alternate way along C. We infer the existence of a subdivision of $K_{3,3}$ in G. Right, X and Y share three vertices. We infer the existence of a subdivision of K_5 in G.

in Y that alternate along C. Otherwise, G would contain a subdivision of a forbidden graph as illustrated in Figure 2.3. We can obtain a convex embedding of G from the convex embedding of G' as follows: place x at the position of z and insert y close to x in the face of $G'-E_y$ incident to z and Y, where $E_y := \{zv \mid v \in Y\}$. We next connect x and y with line segments to their respective neighbors in X and Y, and finally x to y. The previous claim implies that the resulting straight line drawing of G is an embedding. It can easily be made convex using the fact that small perturbations of the vertices of a convex polygon leave the polygon convex. □

The next lemma allows to extend the proposition to graphs that are not necessarily 3-connected and thus concludes the proof of Kuratowski's theorem.

Lemma 2.2.5. *Let G be a graph with at least four vertices, containing no subdivision of K_5 or $K_{3,3}$ and such that the addition of any edge between non-adjacent vertices creates such a subdivision. Then G is 3-connected.*

PROOF. We argue by induction on the number n of vertices of G. Note that for $n = 4$ the lemma just says that K_4 is 3-connected. Since removing an edge in K_5 leaves a 3-connected planar graph, the lemma is also true for $n = 5$. We now assume $n \geq 6$. We claim that G is 2-connected. Otherwise, we could write $G = G_1 \cup G_2$ where G_1 and G_2 have a single common vertex x. Let $y_i \in G_i$, $i = 1, 2$, be adjacent to x. Adding the edge y_1y_2 creates a subdivision K of a forbidden graph. Since K_5 and $K_{3,3}$ are 3-connected and since x and y_1y_2 are the only connections between G_1 and G_2, the vertices of K of degree ≥ 3 must lie all in G_1 or all in G_2. Moreover, K must contain a path using both x and the edge y_1y_2. The subpath between x and the edge y_1y_2 can be replaced by one of the two edges xy_1 or xy_2 to produce another subdivision of the same forbidden graph that does not use y_1y_2, hence contained in G. This last contradiction proves the claim.

Suppose that G has two vertices x, y such that $G-\{x, y\}$ is not connected. We claim that xy is an edge of G. Otherwise, we could write $G = G_1 \cup G_2$ where G_1 and G_2 are connected and only have the vertices x, y in common. $G \cup xy$ must contain a subdivision K of a forbidden graph. As above, the degree three vertices of K must all lie in the same subgraph, say G_1. We could then replace the edge xy in K with a path connecting x and y in G_2 to produce a subdivision of a forbidden graph contained in G_1. We again reach a contradiction.

We now assume for a contradiction that G is not 3-connected and we let x, y be two vertices disconnecting G. By the previous claim, we may write $G = G_1 \cup G_2$ where

$G_1 \cap G_2$ reduces to the edge xy. By the same type of arguments used in the above claims we see that adding an edge to G_i $(i = 1,2)$ creates a subdivision of a forbidden graph in the same G_i. We can thus apply the induction hypothesis and assume that each G_i is 3-connected, or has at most three vertices. By Proposition 2.2.4, both graphs are planar and we can choose a convex embedding for each of them. Let $z_i \neq x, y$ be a vertex of a face F_i of G_i bounded by xy. Note that F_i must be equal to the triangle $z_i xy$. (Otherwise, we could add an edge to G_i inside F_i to obtain a larger planar graph.) Adding the edge $z_1 z_2$ to G creates a subdivision K of a forbidden graph. We shall show that some planar modification of G_1 or G_2 contains a subdivision of a forbidden graph, leading to a contradiction.

If all the vertices of K of degree ≥ 3 were in G_1, we could replace the path of K in $G_2 + z_1 z_2$ that uses $z_1 z_2$ by one of the two edges $z_1 x$ or $z_1 y$. We would get another subdivision of the same forbidden graph in G_1. Likewise, G_2 cannot contain all the vertices of K of degree ≥ 3. Furthermore, $V(G_1) \setminus \{x, y\}$ and $V(G_2) \setminus \{x, y\}$ cannot both contain two vertices of degree ≥ 3 in K since there would be four independent paths between them, although G_1 and G_2 are only connected through x, y and $z_1 z_2$ in $G + z_1 z_2$. For the same reason, K cannot be a subdivision of K_5. Hence, K is a subdivision of $K_{3,3}$ and five of its degree three vertices are in the same G_i. Adding a point p inside F_i and drawing the three line segments px, py, pz_i we would obtain a planar embedding of $G_i + \{px, py, pz_i\}$ that contains a subdivision of $K_{3,3}$. This last contradiction concludes the proof. □

Corollary 2.2.6. *Every triangulation of the sphere with at least four vertices is 3-connected.*

PROOF. By Euler's formula it is seen that such a triangulation has a maximal number of edges. By the previous lemma, it must be 3-connected. □

We end this section with a simple characterization of the faces of a 3-connected planar graph. A cycle of a graph G is **induced** if it is induced by its vertices, or equivalently if it has no chord in G. It is **separating** if the removal of its vertices disconnects G. The set of boundary edges of a face of a plane embedding is called a **facial cycle**.

Proposition 2.2.7. *The face boundaries of a 3-connected plane graph are its non-separating induced cycles.*

PROOF. Suppose that C is a non-separating induced cycle of a 3-connected plane graph G. By the Jordan curve theorem $\mathbb{R}^2 \setminus C$ has two components. Since C is non-separating one of the two components contains no vertices of G. This component is not cut by an edge since C has no chord. It is thus a face of G.

Conversely, consider a face f of G. By Proposition 2.1.4 this face is bounded by a cycle C. If C had a chord $e = xy$ then by the 3-connectivity of G there would be a path p connecting the two components of $C - \{x, y\}$. However, p and e being in the same component of $\mathbb{R}^2 \setminus C$ (other than f), they would cross by an application of θ's lemma 2.1.3. Finally, consider two vertices x, y of $G - C$. They are connected by three independent paths. By θ's lemma f is included in one of the three components cut by these paths and the boundary of this component is included in the corresponding two paths. Hence, C avoids the third path. It follows that $G - C$ is connected. □

This proposition says that a planar 3-connected graph has essentially a unique plane embedding: if we realize the graph as a net of strings there are only two ways of dressing the sphere with this net; they correspond to the two orientations of the sphere.

2.2.2 The Minor Version

A **minor** of a graph G is any graph obtained from a subgraph of G by contracting a subset of its edges. In other words, a minor results from any sequence of contraction of edges, deletion of edges or deletion of vertices (in any order). Equivalently, H is a minor of G if the vertices of H can be put into correspondence with the trees of a forest in G and if every edge of H corresponds to a pair of trees connected by a (non-tree) edge (but all such pairs do not necessarily give rise to edges). Being a minor of another graph defines a partial order on the set of graphs. This partial order is the object of the famous graph minor theory developed by Robertson and Seymour and culminating in the proof of Wagner's conjecture that the minor relation is a **well-quasi-order**, *i.e.* that every infinite sequence of graphs contains two graphs such that the first appearing in the sequence is a minor of the other. As an easy consequence, every **minor closed** family of graphs is characterized by a *finite* set of **excluded minors**. In other words, if a family of graphs contains all the minors of its graphs, then a graph is in the family if and only if none of its minors belongs to a certain finite set of graphs. The set of all planar graphs is the archetypal instance of a minor closed family. Its set of excluded minors happens to be precisely the two Kuratowski graphs.

Theorem 2.2.8 (Wagner, 1937)**.** *A graph G is planar if and only if none of K_5 or $K_{3,3}$ is a minor of G.*

Remark that if G contains a subdivision of H, then H is a minor of G, but the converse is not true in general (think of a counter-example). We can nonetheless deduce Wagner's version from Kuratowski's theorem: the condition in Wagner's theorem is obviously necessary by noting that a minor of a planar graph is planar and by Theorem 2.1.7. The condition is also sufficient by the above remark and by Kuratowski's theorem. In fact, the equivalence between Wagner and Kuratowski's theorems can be shown by proving that a graph contains a subdivision of K_5 or $K_{3,3}$ if and only if K_5 or $K_{3,3}$ is a minor of this graph [Die05, Sec. 4.4].

2.3 Other Planarity Characterizations

We give some other planarity criteria demonstrating the fascinating interplay between Topology, Combinatorics and Algebra.

An **algebraic cycle** of a graph G is any subset of its edges that induces an **Eulerian subgraph**, *i.e.* a subgraph of G with vertices of even degrees[2]. It is a simple exercise to prove that any algebraic cycle can be decomposed into a set of (simple) cycles in the usual acception. The set of (algebraic) cycles is given a group structure by defining the sum of two cycles as the symmetric difference of their edge sets. It can be considered

[2] An Eulerian subgraph in this sense is not necessarily connected.

as a vector space over the field $\mathbb{Z}/2\mathbb{Z}$ and is called the **cycle space**, denoted by $Z(G)$ (the letter Z is short for the German word for cycle, *Zyklus*). The cycle space of a tree is trivial. Also, the cycle space is the direct sum of the cycle spaces of the connected components of G. Given a spanning tree of G, each non-tree edge gives rise to a cycle by joining its endpoints by a path in the tree. It is not hard to prove that these cycles form a basis of the cycle space. Hence, when $G=(V,E)$ is connected,

$$\dim Z(G) = 1 - |V| + |E| \tag{2.1}$$

This number is sometimes called the **cyclomatic number** of G. A basis of the cycle space is a **2-basis** if every edge belongs to at most two cycles of the basis.

Theorem 2.3.1 (MacLane, 1936)**.** *A graph G is planar if and only if $Z(G)$ admits a 2-basis.*

PROOF. It is not hard to prove that a graph that admits a 2-basis has a 2-basis composed of simple cycles only. See Exercise 2.3.2. Such a 2-basis must be the union of the 2-bases of the blocks in the **block decomposition**[3] of G. Moreover, G is planar if and only if its blocks are. We may thus assume that G has a single block, or equivalently that G is 2-connected.

Suppose that G is planar and consider the set B of boundaries of its bounded faces in a plane embedding. Every edge belongs to at most two such boundaries by Proposition 2.1.4. Furthermore, by the same proposition and the Jordan curve theorem, a simple cycle C of G is the sum of the boundaries of the faces included in the bounded region of C. Thus B generates $Z(G)$. Using Euler's formula, the number of bounded faces of G appears to be precisely $\dim Z(G)$. Hence, B is 2-basis.

For the reverse implication, suppose that G has a 2-basis. Note that it is equivalent that any subdivision of G admits a 2-basis. Moreover, $G-e$ has a 2-basis for any edge e: if e appears in two elements of the 2-basis replace these two elements by their sum, otherwise simply remove the basis element that contains e, if any. It follows that any subdivision of a subgraph of G has a 2-basis. We claim that none of the forbidden graphs can have a 2-basis, so that G is planar by Kuratowski's theorem. Indeed, assume the converse and let $C_1,\dots,C_d$ be a 2-basis of a forbidden graph. The C_i's being linearly independent, $\sum_i C_i$ is non-trivial hence contains at least 3 edges. It follows that $\sum_i |C_i| \le 2|E|-3$. From formula (2.1) we compute $\dim Z(K_{3,3}) = 4$. Since every cycle in a bipartite graph has length at least four, we have $\sum_{1\le i\le 4} |C_i| \ge 4\cdot 4 = 16$, in contradiction with $\sum_i |C_i| \le 2\cdot 9-3 = 15$. Similarly, we compute $\dim Z(K_5) = 6$, whence $\sum_{1\le i\le 6} |C_i| \ge 6\cdot 3 = 18$, in contradiction with $\sum_i |C_i| \le 2\cdot 10-3 = 17$. □

Exercise 2.3.2. Show that a graph with a 2-basis admits a 2-basis whose elements are simple cycles. (Hint: Any algebraic cycle is a sum of edge-disjoint simple cycles. Try to minimize the total number of such simple cycles in the 2-basis.)

A **cut** in a graph $G=(V,E)$ is a partition of its vertices. A cut can be associated with the subset of edges with one endpoint in each part. Just as for the cycle space, the set of

[3] The blocks of G are its subgraphs induced by the classes of the following equivalence relation on its set of edges: $e \sim e'$ if there is a cycle in G that contains both e and e'.

cuts can be given a vector space structure over $\mathbb{Z}/2\mathbb{Z}$ by defining the sum of two cuts as the symmetric difference of the associated edge sets. Equivalently, we observe that the sum of two cuts $\{V_1, V_2\}$ and $\{W_1, W_2\}$ is the cut $\{(V_1 \cap W_1) \cup (V_2 \cap W_2), (V_1 \cap W_2) \cup (V_2 \cap W_1)\}$. Remark that the cut space is generated by the **elementary cuts** of the form $\{v, V - v\}$, for $v \in V$. A cut is **minimal** if its edge set is not contained in the edge set of another cut. In a connected graph minimal cuts correspond to partitions both parts of which induce a connected subgraph. Such minimal cuts generate the cut space.

Given a plane graph G, we define its **geometric dual** G^* as the graph obtained by placing a vertex inside each face of G and connecting two such vertices if their faces share an edge in G. Note that distinct plane embeddings of a planar graph may give rise to non-isomorphic duals. When the plane graph G is connected, its vertex, edge and face sets are in 1-1 correspondence with the face, edge and vertex sets of G^* respectively. Note that the geometric dual of a plane tree has a single vertex, so that G^* may not be simple even if G is. It is not hard to prove that the set of edges of a (simple) cycle of G corresponds to a minimal cut in G^*. The converse is also true since the dual of the dual is the original graph.

For non-planar graphs the above construction is meaningless and we define an abstract notion of duality that applies in all cases. A graph G^* is an **abstract dual** of a graph G if the respective edge sets can be put in 1-1 correspondence so that every (simple) cycle in G corresponds to a minimal cut in G^*.

Theorem 2.3.3 (Whitney, 1933)**.** *A graph is planar if and only if is has an abstract dual.*

The theorem can be proved by mimicking the proof of MacLane's theorem 2.3.1, first showing that if a graph has an abstract dual so does its subgraphs and subdivisions. We provide a shorter proof based on MacLane's theorem.

PROOF. The theorem can be easily reduced to the case of connected graphs. By the above discussion a geometric dual *is* an abstract dual, so that the condition is necessary. For the reverse implication, suppose that a graph G has an abstract dual G^*. The cycle space of G is generated by its simple cycles, hence by the dual edge sets of the minimal cuts of G^*. Those cuts are themselves generated by the elementary cuts. Clearly an edge appears in at most two elementary cuts (loop-edges do not appear in any cuts). It follows that the cycle space of G has a 2-basis, and we may conclude with MacLane's theorem. □

We list below some other well-known characterizations of planarity without proof.

A strict **partial order** on a set S is a transitive, antisymmetric and irreflexive binary relation, usually denoted by $<$. Two distinct elements $x, y \in S$ such that either $x < y$ or $y < x$ are said **comparable**. A partial order is a **linear**, or **total**, order when all the elements are pairwise comparable. The **dimension** of a partial order is the minimum number of linear orders whose intersection (as binary relations) is the partial order. The **order complex** of a graph $G = (V, E)$ is the partial order on the set $S = V \cup E$ where the only relations are $v < e$ for v an endpoint of e.

Theorem 2.3.4 (Schnyder, 1989)**.** *A graph is planar if and only if its order complex has dimension at most 3.*

See Mohar and Thomassen [MT01, p. 36] for more details. The **contact graph** of a family of interior disjoint disks in the plane is the graph whose vertices are the disks in the family and whose edges are the pairs of tangent disks.

Theorem 2.3.5 (Koebe-Andreev-Thurston)**.** *A graph is planar if and only if it is the contact graph of a family of disks.*

Section 2.8 in [MT01] is devoted to this theorem and its extensions. A **3-polytope** is an intersection of half-spaces in $\mathbb{R}^3$ which is bounded and has non-empty interior. Its graph, or 1-skeleton, is the graph defined by its vertices and edges.

Theorem 2.3.6 (Steinitz, 1922)**.** *A 3-connected graph is planar if and only if it is the graph of a 3-polytope.*

A proof can be found in the monograph by Ziegler [Zie95, Chap. 4]. We end this section with a nice and simple planarity criterion relying on a result by Hanani (1934) stating that any drawing of K_5 and of $K_{3,3}$ has a pair of independent edges with an odd number of crossings. (Recall that two edges are **independent** if they do not share any endpoint.) In fact, we have the stronger property that the number of pairs of independent edges crossing oddly is odd. This can be proved by first observing the property on a straight line drawing of K_5 (resp. $K_{3,3}$) and then deforming any other drawing to the given one using a sequence of elementary moves that preserve[4] the parity of the number of oddly crossing pairs of independent edges. Together with Kuratowski's theorem, this proves the following

Theorem 2.3.7 (Hanani-Tutte)**.** *A graph is planar if and only if it has a drawing in which every pair of independent edges crosses evenly.*

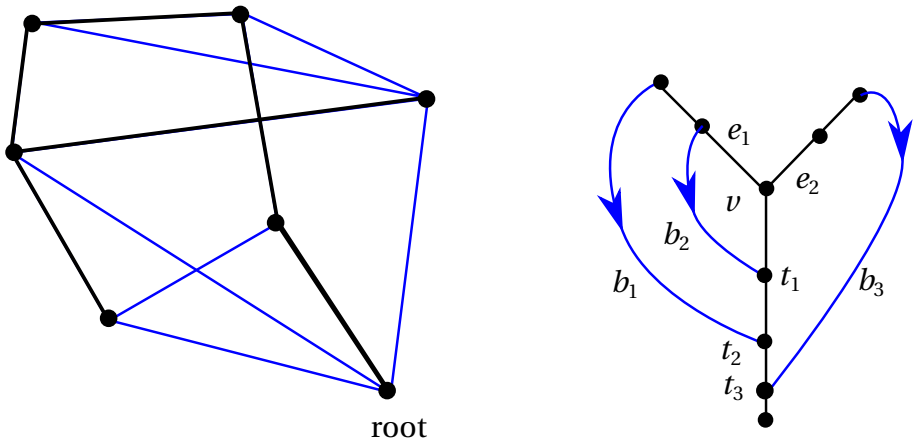
A weaker version of the theorem asks that every pair of edges, not necessarily independent, should cross evenly. See Mustafa's course notes for a geometric proof, not relying on Kuratowski's theorem.

2.4 Planarity Test

There is a long and fascinating story for the design of planarity tests, culminating with the first optimal linear time algorithm by Hopcroft and Tarjan [HT74] in 1974. Patrignani [Pat13] offers a nice and comprehensive survey on planarity testing. Although

[4] Those moves are of five types: (i) two edges locally (un)crossing and creating or canceling a bigon, (ii) an edge locally (un)crossing and creating or canceling a monogon, (iii) an edge passing over a crossing, (iv) an edge passing over a vertex, and (v) two consecutive edges around a vertex swapping their circular order. The three first moves are analogous to the Reidemeister moves performed on knot diagrams. (i),(ii), (iii) and (v) clearly preserve the number of oddly crossing pairs of independent edges. For (iv) we use the fact that for every vertex and every edge of K_5 or $K_{3,3}$ the edge is independent with an even number of the edges incident to the vertex.

most of the linear time algorithms have actual implementations, they are rather complex and we only describe a simpler non optimal algorithm based on works of de Fraissex and Rosensthiel [dFR85, Bra09]. We first recall that the block decomposition decomposes a connected graph into 2-connected subgraphs connected by trees in a tree structure. Hence, a graph is planar if and only if its blocks are planar. We can thus restrict the planarity test to 2-connected graphs. Note that the block decomposition of a graph can be computed in linear time using depth-first search. (See West [Wes01, p. 157].)



a. b.

Figure 2.4: a. A graph (in blue) and a DFS tree in black. b. v is the branching point of a fork, b_1 and b_2 are two return edges for e_1, b_3 is a return edge for e_2 and the lowpoint of e_1 is t_2. The back edges b_1 and b_2 are left and the back edge b_3 is right.

Also recall that a depth-first search in a graph discovers its vertices from a root vertex by following edges that form a spanning tree called a **depth-first search tree**. We say that a vertex v_1 of that tree is **higher** than another vertex v_2 if v_1 is a descendent of v_2. The non-tree edges are called **back edges**. A back edge always connects a vertex to one that is lower in the depth-first search tree. The depth-first search induces an orientation of the tree edges directed from the root toward the leaves of the tree. The back edges are then directed from their highest toward their lowest vertex. Each back edge b defines an oriented **fundamental cycle**, $C(b)$, obtained by connecting its endpoints with the unique tree path between its target and source points. We write uv for an edge directed from u to v. Two fundamental cycles may only intersect along a tree path, in which case the last edge uv along this path together with the **outgoing edges** vw_1 and vw_2 along the two cycles is called a **fork** with **branching point** v. A back edge vw is a **return edge** for itself and for every tree edge xy such that w is lower than x, and v is either higher than y or equal to[5] y. The **return points** of an edge are the targets of its return edges. The **lowpoint** of the edge is its lowest return point, if any, or its source if none exists. The lowpoint of a back edge is thus its target point. We refer to Figure 2.4 for an illustration of all these concepts.

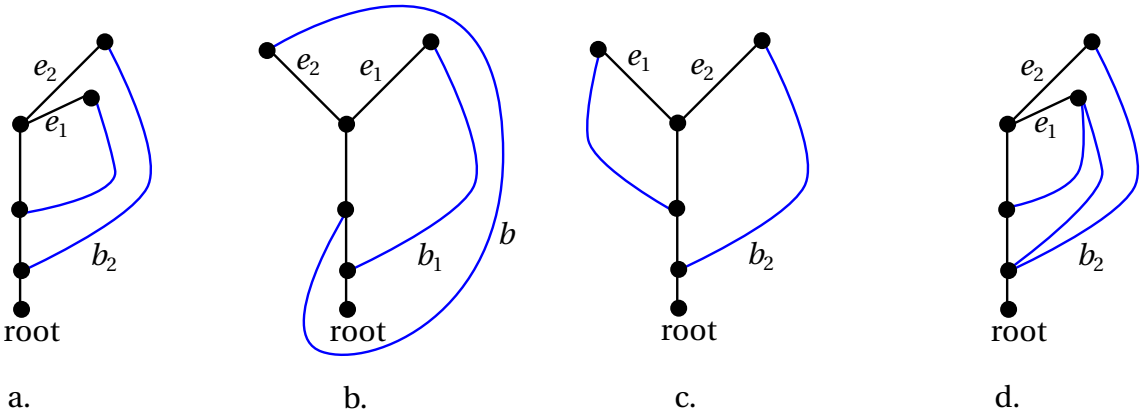
The idea of the planarity test is as follows. Suppose that a graph G has a plane embedding and consider a depth-first search tree of G. Without loss of generality,

[5] a vertex is neither higher nor lower than itself!

we may assume that the root is adjacent to the outer face of the plane embedding. The induced orientation of each fundamental cycle may appear clockwise or counter-clockwise with respect to the embedding of G. A back edge is said **right** (with respect to the embedding) if its fundamental cycle is oriented clockwise, and **left** otherwise. Consider a fork with outgoing edges e_1, e_2. They must have return edges since the graph is 2-connected. Then we have the following necessary conditions:

Fork condition:

1. All return edges of e_1 whose lowpoints are higher than the lowpoint of e_2 have fundamental cycles oriented the same way and
2. all return edges of e_2 whose lowpoints are higher than the lowpoint of e_1 have fundamental cycles oriented the other way.



a. b. c. d.

Figure 2.5: a. and c. The two cases occurring in the fork condition. b. a forbidden case. d. in this case, one chooses $e_2 \prec e_1$.

Lemma 2.4.1. *In a plane embedding, the orientations of the return edges satisfy the fork condition.*

PROOF. Let us denote by b_i the return edge having the same lowpoint as e_i. Then either the disks bounded by $C(b_1)$ and $C(b_2)$ have disjoint interior, or one is included in the other:

- In the latter case, swapping the indices 1 and 2 if necessary, we may assume that e_1 is inside $C(b_2)$. This is pictured in Figure 2.5a. Then any return edge of e_1 must also be inside the disk bounded by $C(b_2)$, and thus be oriented as b_2. In particular, b_1 is oriented as b_2 and e_2 is outside $C(b_1)$. It follows that any return edge of e_2 must lie outside $C(b_1)$. Furthermore, a return edge b from e_2 having lowpoint higher than the one of b_1 must also lie outside $C(b_2)$, since otherwise $C(b)$ could not join its lowpoint without crossing $C(b_1)$. Now, the cycle $C(b)$ cannot contain the root in its interior as on Figure 2.5b, since the root is on the outer face. We infer that b is oriented oppositely to b_2. The fork condition is thus satisfied.

- In the former case, any return edge b of e_1 must lie outside $C(b_2)$. See Figure 2.5c. If the lowpoint of b is higher than that of b_2, then b must be oriented oppositely to b_2, since $C(b)$ cannot contain the root in its interior. The mirror argument shows that a return edge from e_2 whose lowpoint is higher than the lowpoint of b_1 must be oriented oppositely to b_1. We again conclude that the fork condition is satisfied.

□

An **LR partition** is a left-right assignment of the back edges such that the induced orientations of the fundamental cycles satisfy the fork condition for all the possible forks. The above lemma shows that a planar graph has an LR partition deduced from any particular plane embedding. As the following theorem shows, the existence of an LR partition happens to be sufficient for attesting planarity!

Theorem 2.4.2 (de Fraysseix and Rosenstiehl, 1985)**.** *A connected graph G is planar if and only if it admits an LR partition with respect to some (and thus any) depth-first search tree.*

PROOF (SKETCH). Essentially, the proof starts by constructing a combinatorial embedding of G from the LR partition, *i.e.* a circular ordering of the edges around each vertex, then checking that this combinatorial embedding can indeed be realized in the plane without introducing crossings. Note that the fork conditions cannot involve back edges in different blocks in the block decomposition of G, so that we can assume G to be 2-connected by the above discussion. For each vertex v we define a total ordering $\prec$ on its outgoing edges as follows. If v is the root, it can have only a single outgoing edge by the 2-connectivity of G and there is nothing to do. Otherwise, v has a unique incoming tree edge e and the total ordering will correspond to the circular clockwise ordering around v broken at e into a linear ordering. Let e_1, e_2 be two edges going out of v, and for $i = 1, 2$, let b_i be equal to e_i if e_i is a return edge, or a return edge of e_i with the lowest return point (there might be several ones) among its return edges. We need to decide if $e_1 \prec e_2$ or the opposite. The idea is that in any plane drawing of the graph, the ordering of e_1 and e_2 is enforced by the LR-assigment of b_1 and b_2.

- If b_1 is a left back edge while b_2 is a right back edge, then we declare $e_1 \prec e_2$ since it must be the case in any plane drawing of G that respects the LR assignment (as in Figure 2.5c).

- If b_1 and b_2 are both right back edges we let $e_2 \prec e_1$ if either the lowpoint of b_2 is lower than the lowpoint b_1 (as in Figure 2.5a), or if e_1 has another right return edge towards another return point (as in Figure 2.5d). By the fork condition, it is impossible for both e_1 and e_2 to have another right return edge towards another return point, so this is well-defined.

- If b_1 and b_2 are both left back edges, the previous situation leads to the opposite decision.

- If none of this applies, we order them arbitrarily.

There remains to include the incoming return edges in this ordering. Let $e_1 \prec e_2 \prec \cdots \prec e_\ell$ be the resulting ordering of the edges going out of v. We denote by $L(e_i)$ and $R(e_i)$ the left and right incoming back edges whose source points are in the subtree rooted at the target of e_i, or equivalently whose fundamental cycles contain e_i. We order the elements of $L(e_i)$ as follows: we let $b_1 \prec b_2$ if and only if the fork of their cycles $C(b_1)$ and $C(b_2)$ has outgoing edges $a_2 \prec a_1$. An analogous ordering is defined for $R(e_i)$. We finally concatenate all those orderings as follows, the rationale is pictured in Figure 2.6:

$$L(e_1) \prec e_1 \prec R(e_1) \prec L(e_2) \prec e_2 \prec \cdots \prec L(e_\ell) \prec e_\ell \prec R(e_\ell)$$

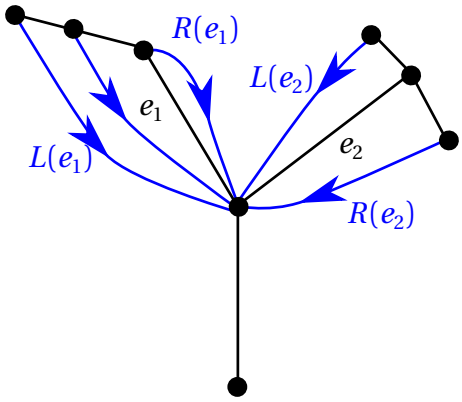


Figure 2.6: Ordering the incoming return edges.

For the root vertex we define the ordering $L(e) \prec e \prec R(e)$ where e is the unique outgoing edge of the root and $L(e), R(e)$ and their ordering are defined similarly as above. It remains to prove that the computed orderings define a planar combinatorial embedding. To this end, we first embed the depth-first search tree into the plane by respecting the computed orderings. This is obviously always possible. We then insert a small initial and final piece for each back edge in its place while respecting the circular orderings and without introducing crossings. Consider a simple closed curve C that goes along the embedding of the depth-first search tree, staying close to it. Each inserted back edge piece intersects C in a single point. Those points are paired according to the back edge to which they belong. We claim that the constructed orderings are such that the list of intersections along C is a well parenthesized expression. To see this we just need to prove that any two pairs of points appear in the good order (not interlaced) along C. There are two cases to consider: the pair corresponds to back edges, say b_1, b_2, that are either on the same side, or on opposite sides. Suppose for instance that b_1 and b_2 are both right edges. If they have the same lowpoint then the constructed orderings implies that their initial and final pieces indeed appear in the good order along C. Similar arguments hold for the other cases. It follows from the claim that we can connect all the paired pieces without introducing crossings, thus proving that G has a plane embedding. □

In order to test if G has an LR partition we can first compute a constraint graph whose nodes are the back edges and whose links are 2-colored constraints: the blue

links connect nodes that must be on the same side and the red links connect nodes that must lie on opposite sides. All the links are obtained from the fork conditions. This graph can easily be constructed in quadratic time with respect to the number of edges of G. It remains to contract the blue links and check if the resulting constraint graph is bipartite to decide if G has an LR partition or not. This can clearly be done in quadratic time.

2.5 Drawing with Straight Lines

Proposition 2.2.4 together with Lemma 2.2.5 show that every planar graph has a straight line embedding. One of the oldest proof of existence of straight line embeddings is credited to Fáry [Fár48] (or Wagner, 1936) and does not rely on Kuratowski's theorem. By adding edges if necessary we can assume given a maximally planar graph G, so that adding any other edge yields a non-planar graph. Every embedding of G is thus a triangulation, since otherwise we could add more edges without breaking the planarity. We show by induction on the number of vertices that any (topological) embedding of G can be realized with straight lines. Choose one embedding. By Euler's formula, G has a vertex v of degree at most 5 that is not a vertex of the unbounded face (triangle) of the embedding. Consider the plane triangulation H obtained from that of G by first deleting v and then adding edges (at most two) to triangulate the face of $G-v$ that contains v in its interior. By the induction hypothesis, H can be realized with straight lines. We now remove the at most two edges that were added and embed v in the resulting face. Since the face is composed of at most 5 edges, it must be star-shaped and we can put v in its center to join it with line segments to the vertices of the face. We obtain this way a straight line embedding of G.

There is another proof of Proposition 2.2.4 due to Tutte [Tut63] that actually provides an algorithm to explicitly compute a convex embedding of any 3-connected planar graph $G=(V,E)$. The algorithm can be interpreted by a physical spring-mass system. Consider a facial cycle C of G (recall that those are determined by Proposition 2.2.7) and nail its vertices in some strictly convex positions onto a plane. Connect every other vertex of G, considered as a punctual mass, to its neighbors by means of springs. Now, relax the system until it reaches the equilibrium. The final position provides a convex embedding! The system equilibrium corresponds to a state with minimal kinetic energy. By differentiating this energy one easily gets a linear system of equations where each *internal* vertex in $V_I := V \setminus V(C)$ is expressed as the barycenter of its neighbors. The barycentric coefficients are the stiffnesses of the springs. In practice, we associate with every edge e in $E \setminus E(C)$ a positive weight (stiffness) λ_e. In fact, if u and v are neighbor vertices it is not necessary that $\lambda_{uv} = \lambda_{vu}$. One may use "oriented" stiffness. Formally, we have

Theorem 2.5.1 (Tutte, 1963)**.** *Every strictly convex embedding of the vertices of C extends to a unique map $\tau : V \to \mathbb{R}^2$ such that for every internal vertex v, its image $\tau(v)$ is the convex combination of the image of its neighbors $N(v)$ with weights λ_{vw}, for $w \in N(v)$:*

$$\forall v \in V_I, \qquad \sum_{w \in N(v)} \lambda_{vw}(\tau(v)-\tau(w)) = 0. \tag{2.2}$$

Moreover, τ induces a convex embedding of G by connecting the images of every pair of neighbor vertices with line segments.

For conciseness, we number the vertices in V_I from 1 to k and the vertices of C from $k+1$ to n (hence, $k=|V_I|$ and $n=|V|$). We also write λ_{ij} for the weight of edge ij and denote by $N(i)$ the set of neighbors of vertex i. We finally put $\lambda_{ij}=0$ for $j\notin N(i)$. We follow the proof from [RG96] and from the course notes of Éric Colin de Verdière `http://www.di.ens.fr/~colin/cours/all-algo-embedded-graphs.pdf`.

Lemma 2.5.2. *If G is connected, the system (2.2) has a unique solution.*

PROOF. (2.2) can be written

$$\Lambda\begin{bmatrix}\tau_1\\ \vdots\\ \tau_k\end{bmatrix}=\begin{bmatrix}\sum_{j>k}\lambda_{1j}\tau_j\\ \vdots\\ \sum_{j>k}\lambda_{kj}\tau_j\end{bmatrix}$$

where τ_i stands for $\tau(i)$ and

$$\Lambda=\begin{bmatrix}\sum_{j=1}^n\lambda_{1j} & -\lambda_{12} & \dots & -\lambda_{1k}\\ -\lambda_{21} & \sum_{j=1}^n\lambda_{2j} & \dots & -\lambda_{2k}\\ \vdots & \vdots & \ddots & \\ -\lambda_{k1} & -\lambda_{k2} & \dots & \sum_{j=1}^n\lambda_{kj}\end{bmatrix}$$

We need to prove that Λ is invertible. Let $x\in\mathbb{R}^k$ such that $\Lambda x=0$ and let x_i be one of its components with maximal absolute value. We set $x_{k+1}=x_{k+2}=\ldots=x_n=0$. Since $(\Lambda x)_i=\sum_{j\in N(i)}\lambda_{ij}(x_i-x_j)=0$ and $\lambda_{ij}>0$ for $j\in N(i)$ we infer that $x_j=x_i$ for $j\in N(i)$. By the connectivity of G, all the x_j, $j=1,\ldots,n$, are null. We conclude that Λ is non-singular. □

In the sequel, we refer to τ as **Tutte's embedding**. We also assume once and for all that G is 3-connected.

Remark 2.5.3. Since the weights are positive the Tutte embedding of every internal vertex is in the relative interior of the convex hull of its neighbors. In particular, this remains true for the projection of the vertex and its neighbors on any affine line.

We shall derive a maximal principal from this simple remark. Let K be a cycle of G. By Proposition 2.2.7, G has a unique embedding on the sphere (up to change of orientation) and its faces can be partitioned into two families corresponding to the two connected components of the complement of K. The vertices of $G-K$ incident to a face in the part that does not contain C are said **interior to** K.

Lemma 2.5.4 (Maximum principle)**.** *Let h be a non-constant affine form over $\mathbb{R}^2$ such that the Tutte embedding of K is included in the half-plane $\{h\le 0\}$ and such that at most two vertices of K are on the line $\{h=0\}$. Then each vertex v interior to K satisfies $h(\tau(v))<0$.*

PROOF. Consider a vertex v interior to K that maximizes h and suppose for a contradiction that $h(\tau(v)) \geq 0$. Let H be the subgraph of G induced by the vertices interior to K and let H_v be the component of v in H. By the above Remark 2.5.3, all the neighbors $w \in N(v)$, which are either interior to K or on K, must satisfy $h(\tau(w)) = h(\tau(v))$. Hence, the Tutte embedding of H_v is included in $\{h \geq 0\}$. Since G is 3-connected, H_v must be attached to K by at least three vertices. These attachment vertices are embedded in $\{h \leq 0\}$ and at least one of them, call it u, is embedded in $\{h < 0\}$ since at most two are on $\{h = 0\}$. Remark 2.5.3 applied to any vertex of H_v adjacent to u then leads to a contradiction. □

Corollary 2.5.5. *The Tutte embedding of every internal vertex lies in the interior of the convex hull of the given strictly convex embedding of C.*

PROOF. By the maximum principle, every half-plane that contains C contains the interior vertices in its interior. □

Let h be a nonzero linear form over $\mathbb{R}^2$. A vertex of G whose Tutte's embedding is aligned with the Tutte embedding of its neighbors in the direction of the kernel of h is said **h-passive**, and **h-active** otherwise.

Lemma 2.5.6. *Let h be a non-trivial linear form and let v be an h-active interior vertex. G contains two paths $U(v,h)$ and $D(v,h)$ such that*

1. *$U(v,h) := v_0, v_1, \dots v_b$ joins $v = v_0$ to a vertex v_b of C and h is strictly increasing along $U(v,h)$, i.e. $h(\tau(v_{j+1})) > h(\tau(v_j))$ for $1 \leq j < b$.*
2. *$D(s,h)$ joins v to a vertex of C and h is strictly decreasing along $D(s,h)$.*

PROOF. Since v is h-active, Remark 2.5.3 implies the existence of some neighbor w with $h(\tau(w)) > h(\tau(v))$. If this neighbor is on C then we may set $U(v,h) = vw$. Otherwise, w is itself h-active and we can repeat the process until we reach a vertex of C, thus defining the path $U(v,h)$. An analogous construction holds for the downward path $D(s,h)$. □

Lemma 2.5.7. *For every non-trivial linear form h, all the interior vertices are h-active.*

PROOF. By way of contradiction, suppose that some interior vertex v is h-passive. By Lemma 2.5.5, some vertex w of C satisfies $h(\tau(w)) > h(\tau(v))$. Since G is 3-connected, we can choose three independent paths P_1, P_2, P_3 from v to w. For $i = 1,2,3$, let Q_i be the initial segment of P_i from v to the first h-active vertex w_i along P_i. Remark that Q_i has at least one edge and that it is contained in the line $\{h = h(\tau(v))\}$. By Lemma 2.5.6, we can choose two paths $U(w_i,h)$ and $D(w_i,h)$ from w_i to vertices on C. By the preceding remark, the three paths Q_i, $U(w_i,h)$ and $D(w_i,h)$ are pairwise disjoint except at w_i. Using that Q_1, Q_2, Q_3 only share their initial vertex v, it is easily seen that $C \cup_{i=1,2,3} (Q_i \cup P(w_i,h) \cup D(w_i,h))$ contains a subdivision of $K_{3,3}$, in contradiction with Kuratowski's theorem. □

Recall that G is supposed to have a plane embedding with facial cycle C. Using the stereographic projection if necessary, we may assume that C is the facial cycle of the unbounded face. We temporarily assume that all facial cycles of G, except possibly C, are triangles.

Lemma 2.5.8. *Let uvx and uvy be the two facial triangles incident to an edge uv of G not in C, then $\tau(x)$ and $\tau(y)$ are on either sides of any line through $\tau(u)$ and $\tau(v)$.*

PROOF. Note that we do not assume $\tau(v) \neq \tau(u)$ in the lemma. Let h be a linear form whose kernel has the direction of a line ℓ through $\tau(u)$ and $\tau(v)$. By Lemma 2.5.6, there are two paths $U(u,h)$ and $U(v,h)$ embedded strictly above ℓ connecting respectively u and v to C. We can extract from $\{uv\} \cup U(u,h) \cup U(v,h)$ a cycle K above ℓ with only u and v on ℓ. By the maximum principle, all the vertices interior to K are embedded strictly above ℓ. One of the faces uvx and uvy must be contained in the interior of K, so that either $\tau(x)$ or $\tau(y)$ is strictly above ℓ. An analogous argument using $D(u,h)$ and $D(v,h)$ shows that one of $\tau(x)$ or $\tau(y)$ is strictly below ℓ. □

Corollary 2.5.9. *All facial triangles uvw are non-degenerate, i.e. $\tau(u), \tau(v)$ and $\tau(w)$ are pairwise distinct.*

PROOF. By Corollary 2.5.5 all the triangles with an edge in C are non-degenerate. By Lemma 2.5.8 all their adjacent triangles are themselves non-degenerate and by connectivity of the dual graph, all the triangles are non-degenerate. □

Corollary 2.5.10. *If all the facial triangles other than C are triangles the Tutte embedding indeed induces a straight line embedding of G.*

PROOF. Since all the facial triangles are non-degenerate, it is enough to prove that their embeddings have pairwise disjoint interiors. Let p be a point contained in the interior of the embedding of some triangle t. Consider a ray r issued from p that avoids all the embeddings $\tau(V)$ of the vertices of G. This half-line crosses some edge e_0 of $t_0 := t$. By Lemma 2.5.9 the other triangle t_1 incident to e_0 crosses r on the other side of t_0, away from p. In turn, r crosses another edge e_1 of t_1 and we define t_2 as the other incident triangle. This way we define a sequence of interior disjoint triangles $t_0, t_1, \ldots, t_i$ and edges $e_0, e_1, \ldots, e_i$ all crossed by r, each time further away from p until we hit C, *i.e.* e_i belongs to C. Remark that t_i only depends on r as it is the unique triangle incident to the intersection of r and C. Let t' be another triangle that contains p in its interior. It gives rise to another sequence $t'_0 = t', t'_1, \ldots, t'_j$ of triangles crossed by r. By the preceding remark, $t_i = t_j$. Since the preceding triangles are defined unambiguously, we conclude that the two sequences are equal. In particular $t = t'$. □

PROOF OF TUTTE'S THEOREM. This last corollary concludes the proof of Tutte's theorem 2.5.1 when all facial cycles other than C are triangles. When this is not the case, we can triangulate the faces other than C, adding $m-3$ edges in each face of length

m to obtain a planar graph G' with the above property. It is possible to put weights on the edges of G', including those of G, so that the solution for system (2.2) written for G' is the same as for the initial system for G. This is a consequence of Remark 2.5.3 and of the next exercise. By Corollary 2.5.10, Tutte's embedding provides a straight line embedding of G'. Removing the extra edges, we obtain a straight line embedding of G. It remains to observe that each face of this embedding is convex since by Lemma 2.5.7 and Remark 2.5.3, the angle at every vertex of a face is smaller than π. □

Exercise 2.5.11. Let p be a point interior to the convex hull of a finite point set P. Show that p is a convex combination of the points of P with strictly positive coefficients only. (Hint: the convex hull of $P \cup \{p\}$ is star-shaped with respect to p.)

One may wonder whether the barycentric method of Tutte could be extended in three dimensions in order to embed a triangulated 3-ball given a convex embedding of its boundary. However, É. Colin de Verdiére et al. gave counterexamples to such an extension showing that expressing each interior vertex as the barycenter of its neighbors does not always yield an embedding [CdVPV03].

3

Surfaces and Embedded Graphs

Contents

3.1 Surfaces

3.1.1 Surfaces and cellularly embedded graphs

A **surface** is a Hausdorff and second countable topological space that is locally homeomorphic to the plane : that means that every point has a neighborhood homeomorphic to $\mathbb{R}^2$. Recall that a space is **Hausdorff** if every pair of distinct points have disjoint neighborhood and is **second countable** if it admits a countable base of open sets. In this course, we will only deal with compact surfaces, and will generally consider surfaces up to homeomorphism, which is why we say “the sphere” instead of “a sphere”.

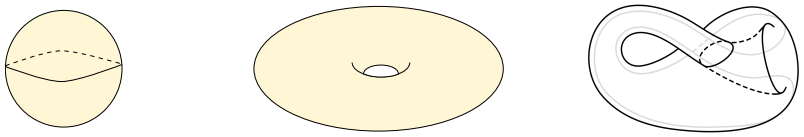


Figure 3.1: The sphere, the torus and the Klein bottle.

Examples of surfaces include the **sphere** $\mathbb{S}^2$, the **torus** $\mathbb{T}^2$, or the **Klein Bottle** $\mathbb{K}^2$, see Figure 3.1. Note that so far, we have not proved that they are different. We emphasize that surfaces are defined intrinsically, i.e., they do not have to be embedded in $\mathbb{R}^3$. For example, the Klein bottle cannot be embedded in $\mathbb{R}^3$: as in Figure 3.1, any representation of it in the usual space induces self-crossings. But this does not prevent it from being a surface : it is behaved locally like the plane which is all that matters here.

Exercise 3.1.1. Consider two copies of the sphere and identify all the corresponding points in the two copies, except for the North pole N. Formally, the resulting space is $\mathbb{S}^2 \times \{0,1\}/\sim$, where $(s,0) \sim (s,1)$ for all $s \in \mathbb{S}^2 \setminus \{N\}$. Show that this space is locally homeomorphic to the plane but that it is not Hausdorff.

Exercise 3.1.2. Show that the plane is second countable. Deduce that a compact space locally homeomorphic to the plane is second countable.

Following our approach outlined in the panorama, we will study surfaces by decomposing them into fundamental pieces, which can be seen as the faces of an embedded graph. Analogously to the planar case, an **embedding** of a graph G into a topological surface Σ is an image of G in Σ where the vertices correspond to distinct points and the edges correspond to simple arcs connecting the image of their endpoints, such that the interior of each arc avoids other vertices and arcs. We first remark that G can always be embedded in some surface. To see this, we can make a drawing of G in the plane and introduce a small handle at every edge intersection as on Figure 3.2 to obtain an embedding.

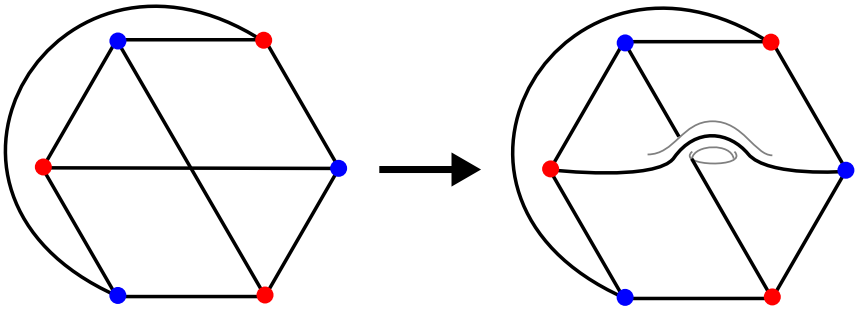


Figure 3.2: A plane drawing of $K_{3,3}$ with a crossing and an embedding in a genus 1 surface.

The **faces** of an embedding are the connected components of $S \setminus G$. A graph is **cellularly embedded** on S if it is embedded and all its faces are homeomorphic to a disk, see Figure 3.3. Thus, describing a cellularly embedded graph amounts to describing a combinatorial way to obtain a surface, by gluing disks together, and one

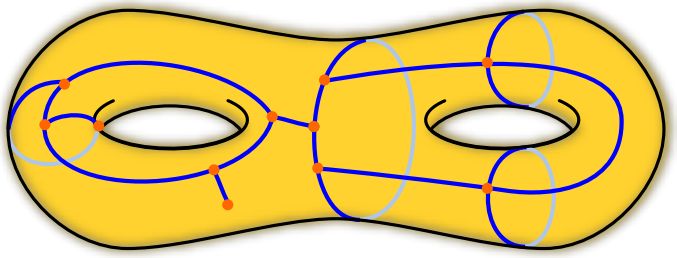


Figure 3.3: The complement of the graph in the surface is a disjoint union of open discs.

can classify surfaces by studying the various possibilities. The following theorem shows that this approach loses no generality:

Theorem 3.1.3 (Kerékjártó-Radó)**.** *On any compact surface, there exists a cellularly embedded graph.*

Since disks can be triangulated, this is equivalent to saying that any compact surface can be triangulated, which is the way this theorem is generally stated in the literature.

PROOF. (Sketch) The result is obvious when the surface, call it S, is a sphere, so we assume this is not the case. Since S is compact and locally planar, it can be covered by a finite number of closed disks D_i, and up to the removal of the superfluous ones, we can assume that no disk lies in the union of any others. Then, if these disks intersect nicely enough (for example if two different boundaries ∂D and $\partial D'$ intersect in a finite number of points), one obtains a finite number of components in $S \setminus \cup \partial D_i$. One can easily show that each of these is a disk (because S is not a sphere!), and therefore one obtains a cellular graph by taking as vertices the intersection points, and as edges the arcs of circles.

So it suffices to show that one can assume that the disks intersect nicely. This can be done by repeated applications of the Jordan-Schoenflies Theorem, but it requires significant work. We refer to Thomassen [Tho92] or Doyle and Moran [DM68] for more details. □

Remark: The issue in this (non-)proof due to a possible infinite number of connected components might look like a mere technicality which one can *obviously* fix. However, we argue that there is a real difficulty lurking there, because the higher dimensional theorem is false: there exists a 4-manifold (A topological space locally homeomorphic to $\mathbb{R}^4$ that can not be triangulated[1], see for example Freedman [Fre82].

As in the planar case, a **triangulation** is a cellular embedding of a graph where all the faces have degree 3. A **subdivision** of a (triangular) face F is obtained by adding a vertex v inside the face, and adding edges between the new vertex v and all the vertices on F, or by adding a vertex w in the middle of an edge and adding edges between w

[1] On a first approximation, it means that it can not be built from a finite number of balls. More formally, it can not be realized as a simplicial complex, which we will introduce later on in the course.

and the non-adjacent vertices in the at most two incident faces. A triangulation is a **refinement** of another triangulation if it is obtained by repeated subdivisions. The same techniques can also be used to prove the following theorem:

Theorem 3.1.4 (Hauptvermutung in 2 dimensions)**.** *Any two triangulations on a given surface have a common refinement.*

We refer to Moise [Moi77] for a proof. As the name indicates ("main hypothesis" in German), this was widely believed to be true *in any dimension*, but once again counterexamples were found in dimensions 4 or higher (see for example [RCS+97]).

3.1.2 Polygonal schemata

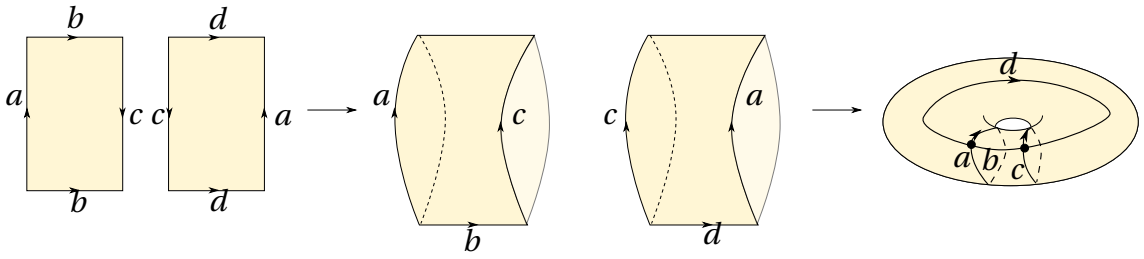


Figure 3.4: From the polygonal scheme $\{a\,b\,c\,\bar{b}, \bar{c}\,d\,\bar{a}\,\bar{d}\}$ to a cellular embedding.

In order to classify surfaces, we introduce **polygonal schemata**, which are a way of encoding the combinatorial data of a cellularly embedded graph : it describes a finite number of polygons with oriented sides identified in pairs. We will see later on, in Section 3.2, other avatars of this combinatorial description of a cellularly embedded graph.

Formally, let S be a finite set of **symbols**, and denote by $\bar{S} = \{\bar{s} \mid s \in S\}$. Then a polygonal scheme is a finite set R of **relations**, each relation being a non-empty word in the alphabet $S \cup \bar{S}$, so that for every $s \in S$, the total number of occurences of s or $\bar{s}$ in R is exactly two.

Starting from a cellularly embedded graph it induces a polygonal scheme in the following way: we first name the edges and orient them arbitrarily. Then for every face, we follow the cyclic list of edges around that face, with a bar if and only if an edge appears in the wrong direction. Every face gives us a relation of R and since every edge is adjacent to exactly two faces, possibly the same, we obtain a polygonal scheme. Conversely, starting from a polygonal scheme, for each relation of size n we build a polygon with n sides, and label its sides following the relation (with the bar indicating the orientation). Then, once all the polygons are built, we can identify the edges labelled with the same label taking the orientations into account. See Figure 3.4.

Exercise 3.1.5. The topological space obtained this way is a compact surface.

Thus, polygonal schemes and cellularly embedded graphs are two facets of the same object. Furthermore, by Theorem 3.1.3, every surface has a cellularly embedded graph, and thus can be obtained by some polygonal scheme. We leverage on this to classify surfaces.

3.1.3 Classification of surfaces

Theorem 3.1.6. *Every compact connected surface is homeomorphic to a surface given by one of the following polygonal schemata, each made of a single relation:*

1. $a\bar{a}$ *(the sphere),*
2. $a_1 b_1 \bar{a}_1 \bar{b}_1 \dots a_g b_g \bar{a}_g b_g$ *for some* $g \geq 1$,
3. $a_1 a_1 \dots a_g a_g$ *for some* $g \geq 1$.

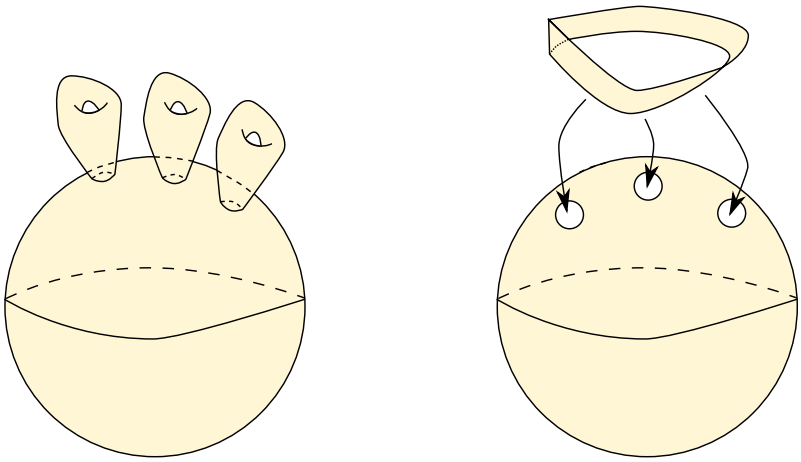


Figure 3.5: The orientable surface of genus 3 and the non-orientable surface of genus 3.

In the second case, the surface is said to be **orientable**, while in the third case it is **non-orientable**. The integer g is called the **genus** of the surface (by convention $g = 0$ for the sphere). In the orientable case, the genus quantifies the number of holes of a surface : an orientable surface of genus g can be built by adding g handles to a sphere. A non-orientable surface of genus g can be built by cutting out g disks of a sphere and gluing g **Möbius bands** along their boundaries. See Figure 3.5.

PROOF. We follow the exposition of Stillwell [Sti93]. Let S be a compact connected surface, and let G be a graph cellularly embedded on S, which exists by Theorem 3.1.3. Whenever an edge of G is adjacent to two different faces, we remove it. Whenever an edge of G is adjacent to two different vertices, we contract[2] it. When this is done, we obtain a cellularly embedded graph G' with a single face and a single vertex. If there are no more edges, then by uncontracting the single vertex into two vertices linked by an edge, we are in case 1 of the theorem and the surface is a sphere. Therefore we can now assume that there is at least one edge.

The graph G' induces a polygonal scheme consisting of a single relation. We will show that this relation can be transformed into either case 2 or case 3 of the theorem without changing the homeomorphism class of S.

[2]The contraction is not meant in the graph-theoretical sense introduced in the earlier chapter : it might result in loops and multiples edges, which we keep.

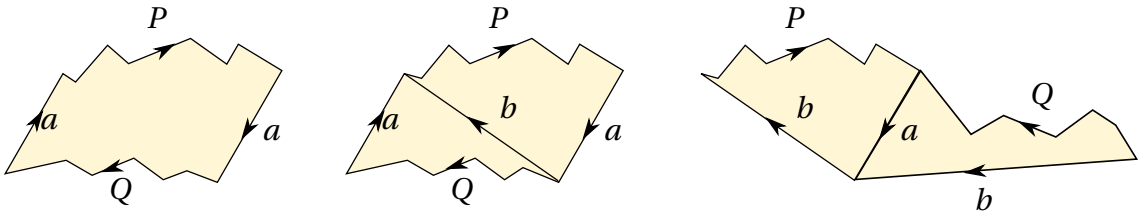


Figure 3.6: From $aPaQ$ to $bbP\bar{Q}$.

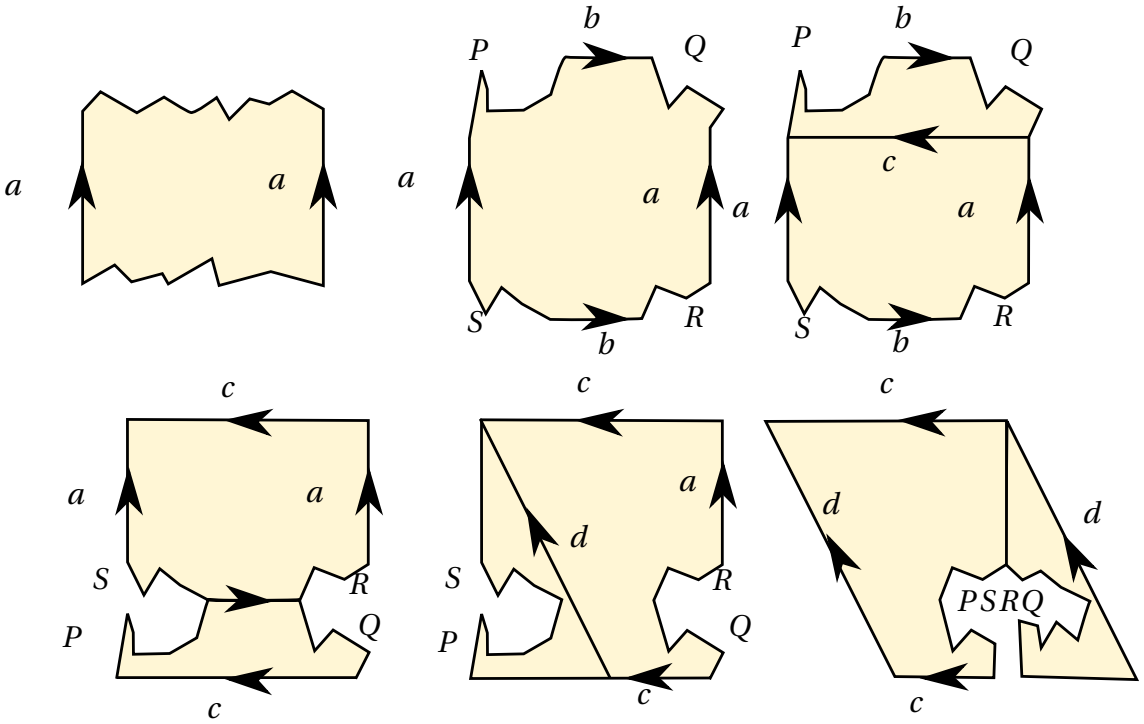


Figure 3.7: From $aPbQ\bar{a}R\bar{b}S$ to $cd\bar{c}\bar{d}PSRQ$.

- If the polygonal scheme has the form $aPaQ$ where P and Q are possibly empty words, then we can transform it into $bbP\bar{Q}$ by adding a new edge and removing a, see Figure 3.6. Inductively, we conclude that each pair of symbols with the same orientation appears consecutively in the polygonal scheme.

- If the polygonal scheme has the form $aU\bar{a}V$, then U and V must share an edge b since otherwise G' would have more than one vertex. By the preceding step, b must appear in opposite orientations in U and V, so we have the form $aU\bar{a}V = aPbQ\bar{a}R\bar{b}S$. This can be transformed into $dc\bar{d}\bar{c}PSRQ$, as pictured in Figure 3.7. Inductively, at the end of this step the relation is a concatenation of blocks of the form aa or $ab\bar{a}\bar{b}$. If all the blocks are of one of these types, we are in case 2 or 3 and we are done.

- Otherwise, the relation has a subword of the form $aabc\bar{b}\bar{c}$. This can be transformed into $d\bar{c}\bar{b}d\bar{b}\bar{c}$, and then using the first step again this can be transformed into $eeffgg$. Inductively, we obtain a relation of the form 3.

This concludes the proof. □

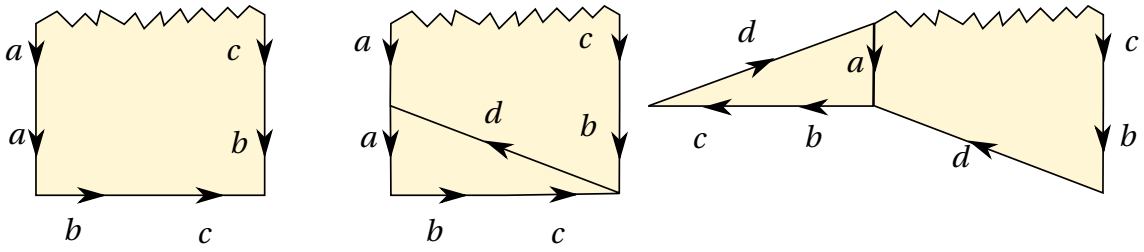


Figure 3.8: From $aabc\bar{b}\bar{c}$ to $\bar{d}\bar{c}\bar{b}\bar{d}\bar{b}\bar{c}$.

Let G be a graph cellularly embedded on a compact surface. The **Euler characteristic** of this embedding equals $v-e+f$, where v is the number of vertices, e is the number of edges and f is the number of faces of the embedded graph.

Lemma 3.1.7. *The Euler characteristic of a graph G cellularly embedded on a surface S only depends on the surface S.*

PROOF. Let G and G' be two cellular embeddings on the same surface S. Since triangulating faces does not change the Euler characteristic, one can suppose that they are triangulated. By Theorem 3.1.4, they have a common refinement. Since subdividing faces does not change the Euler characteristic, this proves the Lemma. □

The Euler characteristic of the surfaces in Theorem 3.1.6 are readily computed from their polygonal schemes: for the sphere we obtain two, for the orientable surfaces $2-2g$ and for the non-orientables ones $2-g$. Therefore the orientable surfaces are all pairwise non-homeomorphic, as are the non-orientable ones. Can orientable surfaces be homeomorphic to non-orientable ones?

Lemma 3.1.8. *A surface S is orientable if and only if it has a cellularly embedded graph G such that the boundary of its faces can be oriented so that each edge gets two opposite orientations by its incident faces.*

PROOF. If the surface S is orientable, then it can be obtained by a polygonal scheme of type 2, for which the boundaries of the faces can be oriented as the lemma requires. If the surface is non-orientable, then any cellularly embedded graph G has a common refinement with one having a polygonal scheme of type 3. Observing that such a graph can not be oriented as the lemma requires, and that this property is maintained when refining, this proves the lemma. □

Corollary 3.1.9. *Orientable surfaces are not homeomorphic to non-orientable ones.*

Therefore, we have established that all the surfaces in Theorem 3.1.6 are pairwise non-homeomorphic. Conversely, any pair of connected surfaces with the same orientability (as defined by Lemma 3.1.8) and Euler characteristic are homeomorphic.

Remark: This classification of surfaces can be extended to the setting of surfaces with boundary: a **surface with boundary** is a topological space where every point is locally homeomorphic to either the plane or the closed half-plane. The **boundary** of such a surface is the set of points that have no neighborhood homeomorphic to the plane. One can show that up to homoemorphism, in line with the above classification, surfaces with boundaries are classified by their genus, their orientability and the number of boundaries (i.e., connected components of the boundary). One way to obtain this is on the one hand to observe that the number of boundaries is a topological invariant, and on the other hand that by gluing disks on the boundaries of a surface with boundary, one obtains a surface without boundary, for which the usual classification applies. The Euler characteristic of the orientable, respectively non-orientable surface of genus g with b boundaries is $2-2g-b$, respectively $2-g-b$.

3.2 Maps

To make things simpler we shall restrict ourselves from now on to orientable surfaces. Up to homeomorphism, a cellular embedding of a graph can be described by the graph itself together with the circular ordering of the edges incident to each vertex. These are purely combinatorial data referred to as a **rotation system**, a **cellular embedding** (of a graph), a **combinatorial surface**, a **combinatorial map**, or just a **map**. The theory of combinatorial maps was developed from the early 1970's, but can be traced back to works of Heffter [Hef91, Hef98] and Edmonds [Edm60] for the combinatorial description of a graph embedded on a surface. The notion of combinatorial map relies on oriented edges rather than just edges. An oriented edge is also called an **arc** or a **half-edge**. Formally, a combinatorial map is a triple (A, ρ, ι) where

- A is a set of arcs,
- $\rho : A \to A$ is a permutation of A,
- $\iota : A \to A$ is a fixed point free involution.

This data allows to recover the embedded graph easily: its vertices correspond to the orbits, or cycles (of the cyclic decomposition), of ρ and its edges correspond to the orbits of ι (so that a and $\iota(a)$ correspond to the two orientations of a same edge). The source vertex of an arc is its ρ-orbit. We shall often write $\bar{a}$ for $\iota(a)$. There are two basic ways of visualizing the corresponding cellular embedding. One way consists in placing disjoint disks in the xy-plane of $\mathbb{R}^3$, one for each vertex, then attaching rectangular strips to the disks, with one strip per edge. The strips should expand in $\mathbb{R}^3$ so that they do not intersect. The counterclockwise ordering of the strips attached to a disc should coincide with the cycle of ρ defining the corresponding vertex. See Figure 3.9 for an illustration. The resulting *ribbon graph* is topologically equivalent to a surface with boundary. Finally glue a disk along each boundary component to a obtain a closed surface where the graph is cellularly embedded. Note that the boundary of each face, traversed with the face to the right, visits the arcs according to the permutation $\varphi := \rho \circ \iota$. The φ-orbits are called **facial walks**. A facial walk need not be simple as can be seen on Figure 3.3. Note that this construction is dual to the concept of polygonal

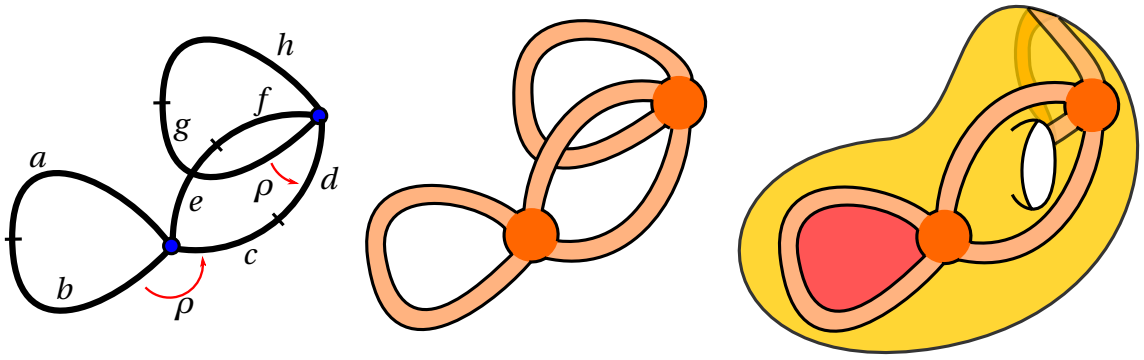


Figure 3.9: A cellular embedding associated to the map (A,ρ,ι) with $A = \{a,b,c,d,e,f,g,h\}$, $\rho = (a,b,c,e)(g,d,h,f)$ and $\iota = (a,b)(c,d)(e,f)(g,h)$. The corresponding graph has a loop edge and a multiple edge.

scheme that we saw earlier : another way of visualizing the cellular embedding is to draw one polygon per facial walk, marking its sides with the arcs of the orbit. Then glue the sides of the polygons that correspond to oppositely oriented arcs (related by the involution ι). Figure 3.10 illustrates this second construction. The numbers

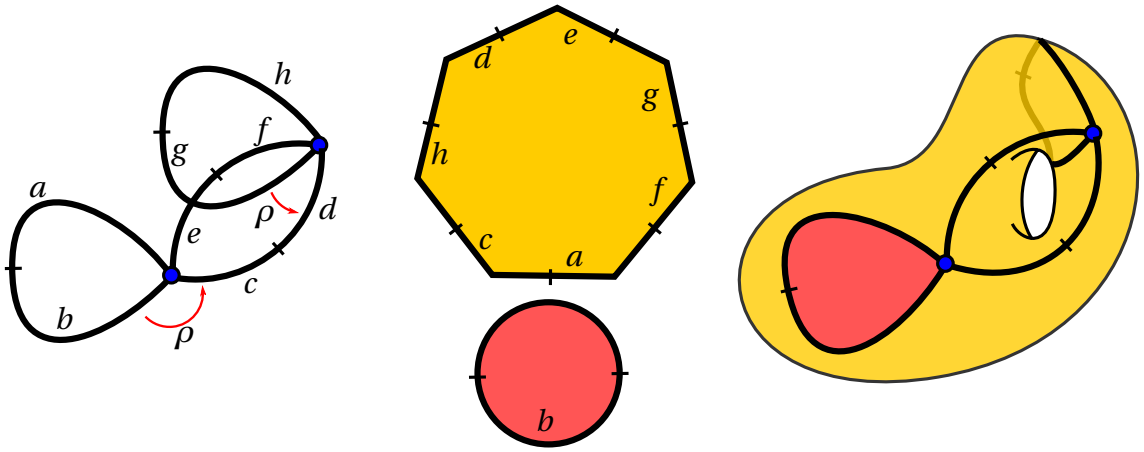


Figure 3.10: Left, the same map as above. Middle, the facial walks of $\varphi = (a,c,h,d,e,g,f)(b)$. Right, the resulting graph embedding.

$|V|,|E|,|F|$ of vertices, edges and faces of the resulting surface are thus given by the number of cycles of the permutations ρ,ι and φ respectively. Obviously, the number of cycles of the involution ι is just $|E| = |A|/2$. The Euler characteristic of this surface can then be computed by the formula

$$\chi = |V| - |E| + |F|.$$

Basic operations on maps. The contraction or deletion of an edge in a graph extend naturally to embedded graphs. Given a map $M = (A,\rho,\iota)$ with graph G, the **contraction** of a non-loop edge $e = \{a,\bar{a}\}$ in G leads to a new map M/e obtained by merging the circular orderings at the two endpoints of e. See Figure 3.11. Formally, $M/e = (A \setminus e, \rho', \iota')$ where ι' is the restriction of ι to $A \setminus e$ and ρ' is obtained by merging

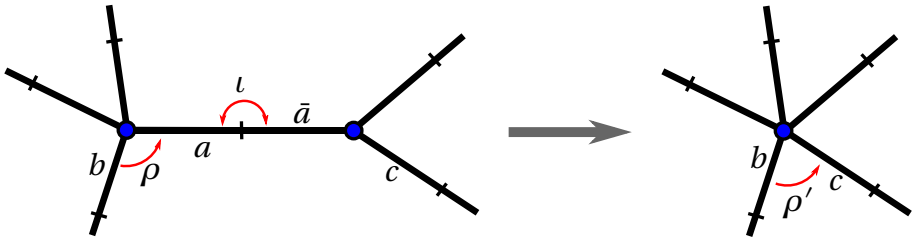


Figure 3.11: The contraction of a non-loop edge. $\rho(b)=a \implies \rho'(b)=\rho\circ\iota(\rho(b))=c$.

the cycles of a and $\bar{a}$, i.e.,

$$\forall b\in A\setminus e,\ \rho'(b)=\begin{cases}\rho(b) & \text{if } \rho(b)\notin e,\\ \rho\circ\iota(\rho(b)) & \text{if } \rho(b)\in e \text{ and } \rho\circ\iota(\rho(b))\notin e,\\ (\rho\circ\iota)^2(\rho(b)) & \text{otherwise.}\end{cases}$$

Likewise, if e has no degree one vertex, the **deletion** of e in G leads to new map $M-e=(A\setminus e,\rho',\iota')$ where ι' is the restriction of ι to $A\setminus e$ and ρ' is obtained by deleting a and $\bar{a}$ in the cycles of ρ, *i.e.*,

$$\forall b\in A\setminus e,\ \rho'(b)=\begin{cases}\rho(b) & \text{if } \rho(b)\notin e,\\ \rho^2(b) & \text{if } \rho(b)\in e \text{ and } \rho^2(b)\notin e,\\ \rho^3(b) & \text{otherwise.}\end{cases} \tag{3.1}$$

Figure 3.12 illustrates the deletion of a loop edge. Let us look at the effect of an edge

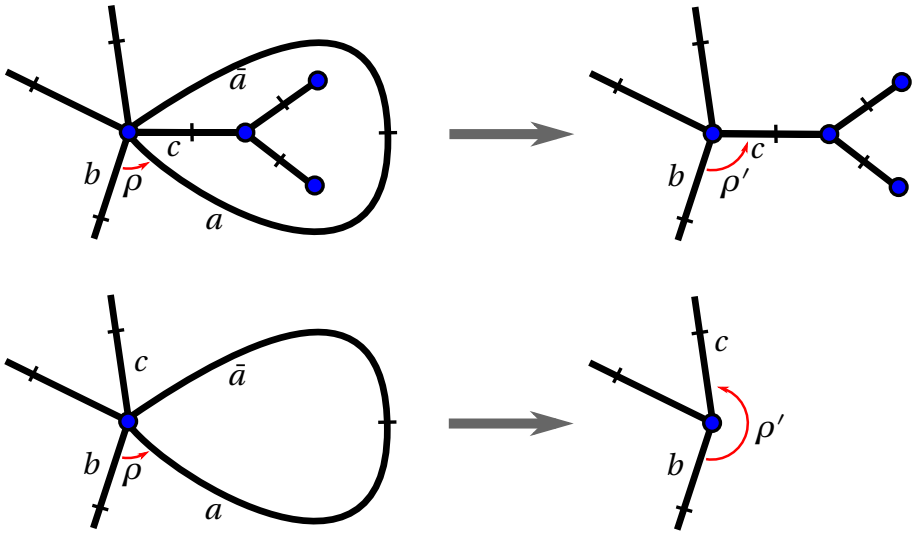


Figure 3.12: The deletion of a loop edge. Above, We have $\rho^2(b)\notin\{a,\bar{a}\}$ implying $\rho'(b)=\rho^2(b)=c$. Below, $\rho^2(b)\in\{a,\bar{a}\}$ so that $\rho'(b)=\rho^3(b)=c$.

contraction or deletion on the topology of a cellular embedding.

Lemma 3.2.1. *If M is a connected map with at least two edges and $e=\{a,\bar{a}\}$ is a non-loop edge of M then M/e is connected and has the same Euler characteristic as M.*

PROOF. The lemma is quite clear if one remarks that M/e has the same number of faces as M but has one edge less and one vertex less than M. Hence,

$$\chi(M \setminus e) = (|V(M)|-1)-(|E(M)|-1)+|F(M)| = \chi(M)$$

□

An edge of an embedded graph is said **regular** if is it incident to two distinct faces and **singular** otherwise.

Lemma 3.2.2. *Let e be an edge of a map M with at least two edges. If e has no vertex of degree one, then*

$$\chi(M-e) = \begin{cases} \chi(M) & \text{if } e \text{ is regular} \\ \chi(M)+2 & \text{otherwise.} \end{cases}$$

Note that the deletion of e may disconnect the map.

PROOF. Clearly, M' has the same number of vertices has M and one edge less. Let $\varphi = \rho \circ \iota$ and $\varphi' = \rho' \circ \iota'$ be the facial permutation of M and M' respectively. Writing $e = \{a, \bar{a}\}$, we have that e is regular if and only if the φ-cycles of a and $\bar{a}$ are distinct. Using formula (3.1), we see that the cycles of φ' are the same as for φ except for those containing a and $\bar{a}$, which are merged if they are distinct for φ and which is split otherwise. We infer that M' has one face less in the former case and one more in the latter. We conclude that

$$\chi(M-e) = |V(M)|-(|E(M)|-1)+(|F(M)|-1) = \chi(M)$$

if e is regular and

$$\chi(M-e) = |V(M)|-(|E(M)|-1)+(|F(M)|+1) = \chi(M)+2$$

otherwise. □

We also define an **edge subdivision** in a map by introducing a vertex in the middle of one of its edges. Likewise, a **face subdivision** consists in the splitting of a face by the insertion of an edge between two vertices of its facial walk. Remark that by contracting one of the two new edges in an edge subdivision one recovers the original map. Similarly, the new edge in a face subdivision is regular and its deletion leads to the original map. It follows from Lemmas 3.2.1 and 3.2.2 that any subdivision of a map preserves the characteristic. Define the **genus of a graph** as the minimum genus of any orientable surface where the graph embeds.

Corollary 3.2.3. *The genus of (a subdivision of) a minor of a graph G is at most the genus of G.*

PROOF. Let M be a cellular embedding of G with minimal genus g. Any subdivision H of a minor of G can be obtained by a succession of edge contractions, deletions and subdivisions. We can perform the same operations on M. By Lemmas 3.2.1 and 3.2.2 and the preceding discussion, the characteristic may only increase during these operations. It follows that the resulting embedding of H has genus at most g, implying that the genus of H is at most g. □

3.3 The Genus of a Map

Thanks to Euler's formula it is quite easy to recover the genus of a map given by a triple (A, ρ, ι). We have:

$$g = 1 - \chi/2 = 1 - (|V| - |E| + |F|)/2 \tag{3.2}$$

where V, E, F are the set of vertices, edges and faces of the map. For a graph G, a certificate that it can be embedded in a surface of genus at most g may be given in the form of a rotation system for G, checking that the genus of the resulting map is at most g. In particular, a graph is planar if and only if it admits a rotation system of genus zero. It follows from the above certificate that computing the genus of a graph is an NP problem. A greedy approach to compute the genus of G is to compute the minimum genus of every possible rotation system for G. For a vertex v the number of possible circular orderings of the incident arcs is $(d_v - 1)!$ where d_v is the degree of v in G. It ensues that the greedy approach needs to consider as much as $\prod_{v \in V(G)} (d_v - 1)!$ rotation systems. It appears that the problem is hard to solve. Indeed,

Theorem 3.3.1 (Thomassen, 1993)**.** *The graph genus problem is NP-complete.*

We first remark that we can restrict the problem to connected simple graphs with at least three vertices. Moreover, given such a graph G, the existence of a rotation system on G that triangulates a surface, *i.e.* such that every facial walk has length three, reduces to the genus problem. Indeed, the number of vertices and edges being fixed by G, only the number of faces may vary among rotation systems. Since the faces of a map correspond to its φ-cycles and since every face has length at least 3, we have $3|F| \leq |A| = |E|/2$ with equality if and only if the map is a triangulation. Formula (3.2) shows that g is minimal in this case. In other words, we can directly deduce from its genus whether G triangulates a surface or not. It is thus enough to show the NP hardness of the triangulation problem. The proof relies on a reduction of the following problem to the triangulation problem.

Proposition 3.3.2 ([Tho93])**.** *Deciding whether a cubic bipartite graph contains two Hamiltonian cycles intersecting in a perfect matching is an NP complete problem.*

Recall that a graph is **cubic** if all its vertices have degree three and it is **bipartite** if its vertices can be split into two sets such that no edge joins two vertices in a same set. A cycle in the graph is **Hamiltonian** if it goes through all the vertices. Finally, a **perfect matching** is a subset of edges such that every vertex is incident to exactly one edge in the subset.

PROOF OF THEOREM 3.3.1. We shall reduce the problem of Proposition 3.3.2 to the triangulation problem. By Proposition 3.3.2 and the discussion after the theorem, the claim implies that the genus problem is NP hard, hence NP complete as we already know it is in NP. Let G be a cubic bipartite graph with a bipartition $A \cup B$ of its vertex set. We construct another graph by first taking a copy G' of G, adding one edge between each vertex v of G and its copy v' in G'. We further add four vertices v_1, v_2, v_1', v_2' and

join v_1 and v_2 to every vertex in G and similarly join v_1' and v_2' to every vertex in G'. Let H be the resulting graph. We next construct a graph Q by contracting all the edges of H of the form $v\,v'$ with $v \in A$ and v' its copy in G'. We claim that
Q triangulates a surface if and only if G admits two Hamiltonian cycles intersecting in a perfect matching.
We first prove the direct implication in the claim, assuming that Q triangulates a surface. In other words there is map with graph Q all of whose faces are triangles. The local rotation of this map around v_1 directly provides a Hamiltonian cycle C_1 in G, which is the boundary of the union of the triangles incident to v_1. Note that every vertex of C_1 is incident to three edges in this union: two edges along C_1 and one edge toward v_1. Similarly, the local rotation around v_2 provides a Hamiltonian cycle C_2. If C_1 and C_2 had two consecutive edges in common, then their shared endpoint would be incident to exactly two more edges: one toward v_1 and one toward v_2. However, by construction v is also incident to its copy in G' or to v_1' and v_2' if $v \in A$, leading to a contradiction. Moreover, since G is cubic C_2 cannot miss two consecutive edges of C_1. It follows that C_1 and C_2 share one of every two edges, hence a perfect matching.

To prove the reverse implication of the claim, suppose now that G has two Hamiltonian cycles C_1 and C_2 intersecting in a perfect matching. We consider the following rotation system on H. If $v \in A$, we let e_i, for $\{i, j\} = \{1, 2\}$, be the arc of $C_i \setminus C_j$ with source vertex v and we let e_3 be the third incident arc, hence in $C_1 \cap C_2$. The local rotation around v is then given by the cycle

$$(e_1, v\,v_1, e_3, v\,v_2, e_2, v\,v')$$

The local rotation around a vertex in $V(G') \setminus A'$ is defined analogously and so is the local rotation around a vertex in A' or in $V(G) \setminus A$, except that we exchange the indices 1 and 2 in the above cycle. See Figure 3.13. For $\{i, j\} = \{1, 2\}$, we define the local rotation

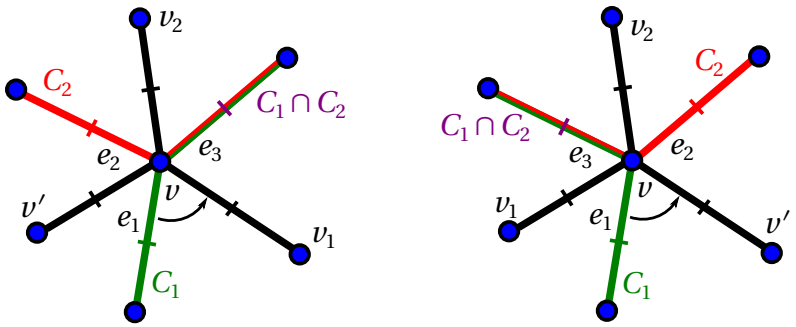


Figure 3.13: Left, the local rotation at a vertex $v \in A \cup V(G') \setminus A'$. Right, the local rotation at a vertex $v \in A' \cup V(G) \setminus A$. The vertices v, v' are copies of the same vertex in G and G'.

around v_i as the cycle C_i oriented so that the edges in common with C_j (previously denoted by e_3) are directed from $V(G) \setminus A$ to A for $i = 1$ and from A to $V(G) \setminus A$ for $i = 2$. Similarly, we define the local rotation around v_i' as the cycle C_i' oriented oppositely to C_i, *i.e.* so that the edges in common with C_j' are oriented from A to $V(G) \setminus A$ when $i = 1$ and from $V(G') \setminus A'$ to A' when $i = 2$. It is an exercise to check that the facial walks of the resulting rotation system have the following form, where $\{i, j\} = \{1, 2\}$:

- A triangle defined by v_i and some edge of C_i, or
- A triangle defined by v'_i and some edge of C'_i, or
- a quadrilateral defined by an edge of $C_i \setminus C_j$, its copy in G' and the two edges between their endpoints from G to G'.

Figure 3.14 shows some of those facial walks. Hence, by contracting the edges $v\,v'$ of

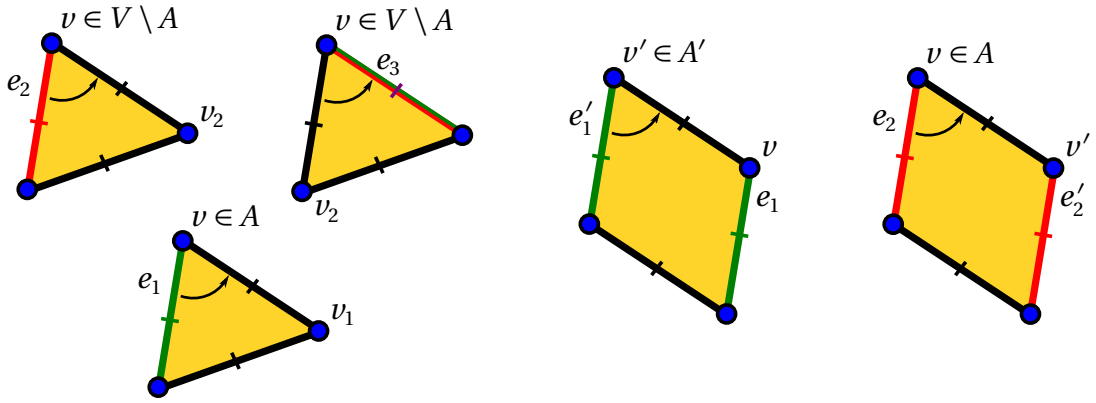


Figure 3.14: Some faces of the rotation system for H.

H with $v \in A$ (and its copy $v' \in A'$), the quadrilaterals are transformed into triangles and one obtain a triangular embedding for Q. □

It can be shown that the graph genus problem is fixed parameter tractable (FPT) with respect to the genus. More precisely, the question whether a graph of size n has genus at most g can be answered in $O(f(g)n)$ time where $f(g)$ is some singly exponential function of g [Moh99, KMR08]. Interestingly, the genus of the complete graphs are known. It was conjectured by Heawood in 1890 that the genus of the complete graph K_n over $n \geq 3$ vertices is

$$g(K_n) = \left\lceil \frac{(n-3)(n-4)}{12} \right\rceil$$

This conjecture was eventually established in 1968 by Ringel and Youngs. The long proof [Rin74] provides explicit minimal genus embeddings of K_n with a different construction for each residue of n modulo 12.

3.4 Homotopy

In a nutshell, two curves drawn on a surface are homotopic if there exists a continuous deformation between them. This intuitive notion, dating back to Poincaré, naturally leads to a very rich theory drawing a bridge between topology on one side and group theory on the other side. We start by introducing the relevant background in group theory.

3.4.1 Groups, generators and relations

Although most groups we will be dealing with in this course are infinite, they can often be very succinctly encoded in terms of **generators** and **relations**. The attentive reader will probably notice some similarities between the formalism established here and the definitions on polygonal schemata : as we will see later on, this is no coincidence.

Let us consider a set G of **generators**, and denote by G^{-1} their **inverses** $G^{-1} = \{g^{-1} \mid g \in G\}$. A word is a string over the alphabet $G \cup G^{-1}$, and we denote by ε the empty word. We consider that two words are equivalent if one can switch from one to the other by adding or removing words of the form gg^{-1} or $g^{-1}g$. The set of finite words quotiented by this equivalence relation can naturally be endowed with the structure of a group: the law is the concatenation, and the neutral element is ε. Indeed:

- The concatenation is well-defined with respect to the equivalence relation and is associative.
- For any word w, $w\varepsilon = \varepsilon w = w$.
- Each element has an inverse obtained by reversing the order of the letters and inverting them, e.g., $c^{-1}b^{-1}a^{-1}$ is the inverse of abc.

This group is called the **free group** on the set G, denoted by $F(G)$.

Now, let us also fix a set R of words called the **relations**, and consider the free group $F(G)$, where one also identifies a word with the word obtained by inserting at any place a word taken from R or their inverses. This defines another group, which is formally the quotient of $F(G)$ by the normal subgroup generated by R[3]. This group is said to admit the **presentation** $< G \mid R >$. So the free group admits the presentation $< G \mid \emptyset >$, generally abbreviated by $< G >$.

Here are a few examples:

- The group $\mathbb{Z}$ is the free group on one letter $F(\{a\})$.
- The group $< a \mid aa >$ is the group $\mathbb{Z}_2$ (or $\mathbb{Z}/2\mathbb{Z}$).
- The group $< a, b \mid aba^{-1}b^{-1} >$ is the two-dimensional lattice $\mathbb{Z}^2$: indeed, the relation $aba^{-1}b^{-1}$ implies that $ab = ba$, thus the group is abelian, and the isomorphism with $\mathbb{Z}^2$ is the map $a \mapsto (1,0)$, $b \mapsto (0,1)$.

Exercise 3.4.1. Recognize the groups $< a, b \mid aab, ab^{-1}b^{-1} >$ and $< a, b \mid a^m, b^n, aba^{-1}b^{-1} >$.

Exercise 3.4.2. Show that any group admits a presentation (with possibly an infinite number of generators and relations), and that any finite group admits a finite presentation.

3.4.2 Fundamental groups, the combinatorial way

We start by introducing homotopy in a combinatorial setting, which makes computations very convenient. The baby case is the case of graphs, which corresponds directly to free groups.

[3] Recall that a subgroup $H \subseteq G$ is normal if $gH = Hg$ for any $g \in G$.

Fundamental groups of graphs

Let G denote a graph (not necessarily embedded) where the edges are oriented. An **arc** is an oriented edge or its inverse, it has an **origin** $o(a)$ and a **target** $t(a) = o(a^{-1})$. A **path** in G is a sequence of arcs $(e_1, \dots, e_n)$ such that the target of e_i coincides with the origin of e_{i+1}. A **loop** is a path such that the target of e_n coincides with the origin of e_1, this point is called the **basepoint** of the loop. The **trivial loop** is the empty loop. Two loops with a common basepoint x are **homotopic** if they can be related to each other by adding or removing subpaths of the type (e, e^{-1}), and a path is **reduced** if it does not contain such a subpath.

Let x be a vertex of G. The set of homotopy classes of loops in G forms a group, where the law is the concatenation and the neutral element the trivial loop. Indeed:

- The concatenation is well-defined with respect ot the equivalence relation and is associative.
- Concatenating with the trivial loop does not change a loop.
- Every loop $(e_1, \dots, e_n)$ has an inverse $(e_n^{-1}, \dots, e_1^{-1})$.

This group is called the **fundamental group** of G, denoted by $\pi_1(G, x)$.

Theorem 3.4.3. *Let T be a spanning tree of G containing the vertex x. Then the fundamental group $\pi_1(G, x)$ is isomorphic to the free group generated by the edges of G that are not in T.*

PROOF. Let C denote the set of edges not in T. For every arc a, one can associate a loop based at x denoted by $\gamma_a^T = \gamma_{x \to o(a)}^T \cdot a \cdot \gamma_{t(a) \to x}^T$, where $\gamma_{x \to y}^T$ denotes the unique reduced path in T between x and y. Then, every loop $(e_1, \dots, e_n)$ based at x in G is homotopic to the loop $\gamma_{e_1}^T, \dots, \gamma_{e_n}^T$, and for every arc a in T, γ_a^T is homotopic to the constant loop. Therefore, $\pi_1(G, x)$ is generated by the loops γ_a^T for a and arc not in T, and since $\gamma_{a^{-1}}^T = (\gamma_a^T)^{-1}$, it is enough to pick one arc for every edge of C. Finally, since the loop γ_a^T for a not in T is the only one containing a, there is no non-trivial relation between the loops γ_a^T. This proves the theorem. □

An alternative way of seeing this proof is to observe that the fundamental group of G is the same as the fundamental group of G obtained after contracting a spanning tree of G. The resulting graph is a bouquet of circles, and there is one generator for each circle.

Fundamental groups of surfaces

Now, let S denote a connected surface and let G be a graph cellularly embedded on S. Similarly as before, a **loop** and a **path** in (S, G) is a path, respectively a loop in G. An **elementary homotopy** between two loops is either a reduction (deletion/addition of $e\,e^{-1}$) or the deletion/addition of a subpath bounding a face of G. This corresponds to the idea that in a continuous deformation between two curves, one can flip a face of a cellularly embedded graph, see Figure 3.15. Elementary homotopies induce

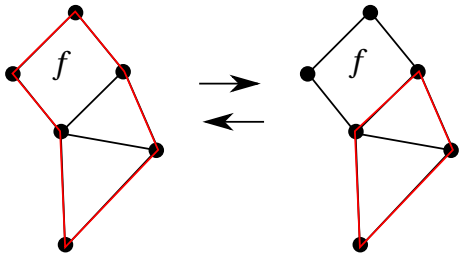


Figure 3.15: The two red paths are homotopic since they are related by going over the face f.

an equivalence relation, called **homotopy** between loops based at a common point, denoted by $\equiv$.

As before, the set of homotopy classes of loops based at a vertex x of G forms a group, where the law is the concatenation. Indeed,

- If $\gamma_1 \equiv \gamma_1'$ and $\gamma_2 \equiv \gamma_2'$ are two pairs of homotopic loops, then their concatenations are homotopic : $\gamma_1\gamma_2 \equiv \gamma_1'\gamma_2'$.
- The concatenation law is associative.
- The trivial loop, denoted by 1_x or simply 1 is a neutral element for the concatenation law.
- Every loop $(e_1, \ldots, e_n)$ admits $(e_n^{-1}, \ldots, e_1^{-1})$ as an inverse.

This group is called the **fundamental group** of S, denoted by $\pi_1(S, x)$[4].

Exercise 3.4.4. Let x and y be two vertices of G, then show that $\pi_1(S, x)$ and $\pi_1(S, y)$ are isomorphic. This justifies the common abuse of notation to just write $\pi_1(S)$ without specifying a base point.

Let T be a spanning tree of G containing a vertex x, and let C denote the set of edges not in T. The fundamental group of G is the free group on C, and to obtain the fundamental group of S from it, one just needs to add the relations corresponding to the faces of G. Formally, for every face $f = (e_1, \ldots, e_n)$ of G, denote by r_f the **facial relation** induced by f on C, that is, the word obtained by only keeping the e_i that are in C. Then we have the following theorem:

Theorem 3.4.5. *Let S be a connected surface, G be a cellularly embedded graph on S with a vertex x and a set of faces F, and T be a spanning tree of G containing x. Denote by C the set of edges not in T and by r_f the facial relation induced by a face f of G on C. Then $\pi_1(S)$ is isomorphic to the group π presented by*

$$< C \mid \{r_f\}_{f \in F} > .$$

[4]To be accurate, we should write $\pi_1(S, G, x)$ but as we will shortly see, this actually does not depend on G.

PROOF. As before, every arc a in G corresponds to a loop γ_a^T obtained by adjoining the reduced paths in T between x and the endpoints of a. Let us consider the map $\gamma : C \to \pi_1(S)$ mapping every arc a in C to γ_a^T. This map induces a morphism of groups $\gamma' : F(C) \to \pi_1(S)$. As for homotopy in graphs, this map is surjective since any loop of S is the image by γ of its arcs in C. We will show that its kernel equals the normal subgroup N generated by the elements r_f for $f \in F$, which proves the theorem.

Let $w = e_1, \dots e_n$ be an element of $F(C)$ such that $\gamma'(w) \equiv 1$. Then it means that $\gamma'(w)$ can be reduced to the trivial loop by a sequence of elementary homotopies. Then for every reduction over edges of C, the same reduction can be applied to w, and for every face flip over a face f, the corresponding facial relation r_f can be used to modify the word w. Thus w can be reduced to the trivial word using the set of relations r_f and the kernel of γ' is included in N. Reciprocally, an element of N is mapped to the trivial loop by γ' since the facial relations r_f dictate the face flips to do to simplify the corresponding relations. This concludes the proof. □

Now, let us observe that the group $\pi(S)$ stays the same when one

1. contracts an edge of G between two different endpoints,
2. or one removes an edge of G between two different faces.

For 1, let e be the edge we are contracting, and T be a spanning tree of G containing e. This contraction yields a new tree T' with one less edge, but the set C of non-tree edges stays the same, as well as the set of facial relations. So the group stays the same.

For 2, when one removes an edge e of G between two different faces, one merges the two adjacent faces f_1 and f_2 into a single face f. One can pick the spanning tree so that e is **not** in it, and thus $\pi_1(S)$ lost one generator g, and the two relations r_1 and r_2 containing g have been merged into one. Observing that this amounts to deleting every appearance of g in $\pi_1(S)$ using r_1 (or r_2), we see that this operation does not change the group.

Therefore, by the classification of surfaces (or rather its proof), we see that the graph G can be transformed into one of the polygonal schemata of Theorem 3.1.6 without changing the fundamental group. In particular, $\pi_1(S)$ only depends on the surface S and not the graph G, and it is isomorphic to

- the trivial group if S is a sphere,
- the group $< a_1, b_1, \dots a_g, b_g \mid a_1 b_1 \bar{a}_1 \bar{b}_1 \dots a_g b_g \bar{a}_g b_g >$ if S is the orientable surface of genus g,
- or $< a_1, \dots, a_g \mid a_1 a_1 \dots a_g a_g >$ if S is the non-orientable surface of genus g.

Remark: The operations of contraction and deletion of edges used above can be interpreted in the light of **dual graphs**: a graph G embedded on a surface S has a dual graph G^*, defined by placing one vertex in each face of G and edges between adjacent faces. Then, contracting an edge in the primal graph amounts to removing an edge in the dual graph, and vice versa. Contracting every edge between different

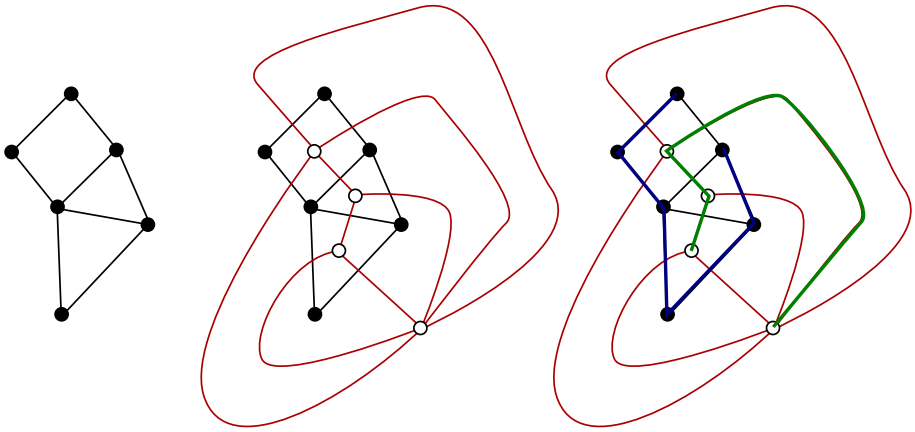


Figure 3.16: A planar graph, its dual graph and a pair of interdigitating spanning trees.

endpoints and removing every edge between different faces amounts to contracting a spanning tree T of G and a spanning tree T^* of G^* which are **interdigitating**, that is, such that the edges of T are not duals of edges of T^*, see Figure 3.16. The use of such interdigitating trees, also called a **tree-cotree decomposition**, is an important tool in the study of embedded graphs, especially from an algorithmic point of view, but we will not rely on it further in this course.

3.4.3 Fundamental groups, the topological way

The homotopies we defined in the previous section are very combinatorial, and do not match our a priori intuition of a continuous deformation. In this section, we define homotopies in a topological way, and show that the corresponding notion of fundamental group matches the one obtained before.

In a purely topological setting, we are now only considering a surface S, without any mention of a cellularly embedded graph. A **path** on S is a continuous map $p : [0,1] \to S$, and a **loop** based at x is a path $p : [0,1] \to S$ where $p(0) = p(1) = x$. A **homotopy** with basepoint x between two loops ℓ_1 and ℓ_2 is a continuous map $h : [0,1] \to [0,1] \to S$ such that $h(0,\cdot) = \ell_1$, $h(1,\cdot) = \ell_2$ and $h(\cdot,0) = h(\cdot,1) = x$. The **constant loop** at x is the loop $p : [0,1] \mapsto x$. The **inverse** of a loop ℓ^{-1} is defined by $\ell^{-1}(t) = \ell(1-t)$. The **homotopy class** of a loop is the set of loops homotopic to it.

The **concatenation** of two loops ℓ_1 and ℓ_2 is the loop defined by $\ell_1(2t)$ for $t \in [0,1/2]$ and $\ell_2(2t-1)$ for $t \in [1/2,1]$. The set of homotopy classes of loops based at x forms a group for the concatenation law, where the neutral element is the constant loop. Indeed:

- If $\gamma_1 \equiv \gamma_1'$ and $\gamma_2 \equiv \gamma_2'$ are two pairs of homotopic loops, then their concatenations are homotopic : $\gamma_1\gamma_2 \equiv \gamma_1'\gamma_2'$.
- The concatenation law is associative.
- The constant loop, denoted by 1_x or simply 1 is a neutral element for the concatenation law.
- The inverse of a loop is its inverse for the concatenation law.

This group is called the **fundamental group** $\pi_1(S,x)$.

Remark: We are always working with loops with basepoints, and the homotopies preserve this basepoint. This is needed to obtain a nice algebraic structure: otherwise there is no natural way to concatenate loops. But the study of homotopies without basepoints, called **free homotopies**, is arguably more natural. We will see later on how to include it in this framework.

The following exercise mirrors Exercise 3.4.4:

Exercise 3.4.6. If S is a connected surface, then for every $(x,y)\in S$, the groups $\pi_1(S,x)$ and $\pi_1(S,y)$ are isomorphic. This justifies the common abuse of notation to just write $\pi_1(S)$ without specifying a base point.

As the notations suggest, the topological fundamental group and the combinatorial group turn out to be isomorphic, this is the point of the following theorem.

Theorem 3.4.7. *The topological fundamental group is the same as the combinatorial fundamental group.*

PROOF (SKETCH). Just for the time of this proof, let us denote respectively by $\pi_1^{comb}(S)$ and $\pi_1^{top}(S)$ the combinatorial and topological fundamental groups. We pick a cellularly embedded graph G on S, which will be used to study $\pi_1^{comb}(S)$ (but as we saw, the group itself does not depend on G). We will study the map $\varphi : \pi_1^{comb}(s) \to \pi_1^{top}(S)$ mapping a homotopy class of loops to the corresponding topological homotopy class of loops.

Claim 1: The map φ is well-defined and is a morphism of groups.

Indeed, if two combinatorial loops γ_1 and γ_2 are homotopic, then they are related by a sequence of reductions and face flips. Such reductions and face flips can be realized using topological homotopies, so their images $\varphi(\gamma_1)$ and $\varphi(\gamma_2)$ are homotopic. This map behaves nicely under composition laws, so it is a morphism.

Claim 2: The map φ is surjective.

Let γ be a topological loop on S. It suffices to prove that it is homotopic to a combinatorial loop of G. By perturbing by a very local homotopy if needed, one can assume that γ crosses G a finite number of times. Then between each pair of crossings, on can push γ on one side (for example the left one), so that one obtains a homotopic loop lying entirely in G. Now, it may happen that γ backtracks in the middle of an edge of G, but using a homotopy, one can reduce it so that it does not happen, and thus we obtain a combinatorial loop of G.

Claim 3: The map φ is injective.

Let γ be a combinatorial loop such that $\varphi(\gamma)=1$. This means that some topological homotopy contracts $\varphi(\gamma)$ to the empty loop. We want to discretize this homotopy so that it becomes a concatenation of face flips and reductions. To do that, push every loop in the topological homotopy into a combinatorial loop of G using the previous technique. By construction, the difference between two consecutive such loops will be a series of reductions or face flips, and thus we obtain a combinatorial homotopy.

□

Remark: The "map" π_1 associates a group for any surface, and it can furthermore be seen as a **functor** : continuous maps between surfaces also induce morphisms between their fundamental groups: indeed, such a continuous applications maps loops to loops, and by taking their homotopy classes, one obtains a morphism. Such functors are the playground of **category theory**, which has deep connections with algebraic topology, but we will not delve at all into these aspects in this course.

All in all, we have just seen that properties regarding continuous deformations of loops can be rephrased in a purely group-theoretical point of view. This is very fruitful from a conceptual perspective, as it provides a strong algebraic structure to work with. But from an algorithmic perspective, the benefits are not that immediate : the issue is that working with presentations of groups is very unwieldy, as struggling with the following exercise showcases:

Exercise 3.4.8. Show that the fundamental groups of non-homeomorphic surfaces are not isomorphic.

In fact, most computational problems for group presentations are actually **undecidable**. This is the case for example for the following problems:

- Deciding whether two groups provided by a finite presentation are isomorphic.
- Deciding whether a given group provided by a finite presentation is trivial.
- Deciding whether an element in a group provided by a finite presentation is trivial.

We will establish such undecidability results in a later chapter and see that they translate into undecidable topological problems in higher dimensions. Fortunately, fundamental groups of surfaces are simpler than general groups, and thus we will be able to devise algorithms for homotopy on surfaces, but these algorithms will have a very strong geometric or topological appeal, instead of a group theoretical one. More generally, studying groups by realizing them as fundamental groups of some topological space is one of the drives of **combinatorial group theory**.

3.4.4 Covering spaces

A **covering space** of S is a space $\widehat{S}$ together with a continuous surjective map $\pi : \widehat{S} \to S$ such that for every $x \in S$, there exists an open neighborhood U of x such that $\pi^{-1}(U)$ is a disjoint union of homeomorphic copies of U. The reader scared by this definition should look at the example of the annulus on the left of Figure 3.17.

We will only deal with covering spaces in an informal way, and refer to a standard textbook in algebraic topology like Hatcher [Hat02] for more precise statements.

The reason we are interested in covering spaces is that they are deeply connected with homotopy and fundamental groups. Indeed, a covering space allows to **lift** a path: if p is a path on S such that $p(0) = x = \pi(\widehat{x})$ for some $\widehat{x} \in \widehat{S}$, there is a unique path $\widehat{p}$ on $\widehat{S}$ starting at $\widehat{x}$ such that $p = \pi \circ \widehat{p}$. This is pictured in Figure 3.17. Note that loops do not lift necessarily to loops! This "unique lifting property" derives from the

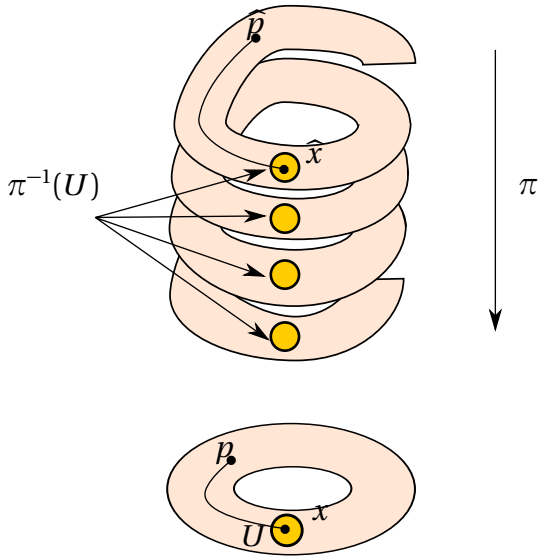


Figure 3.17: A covering of the annulus, and one lift of a path p.

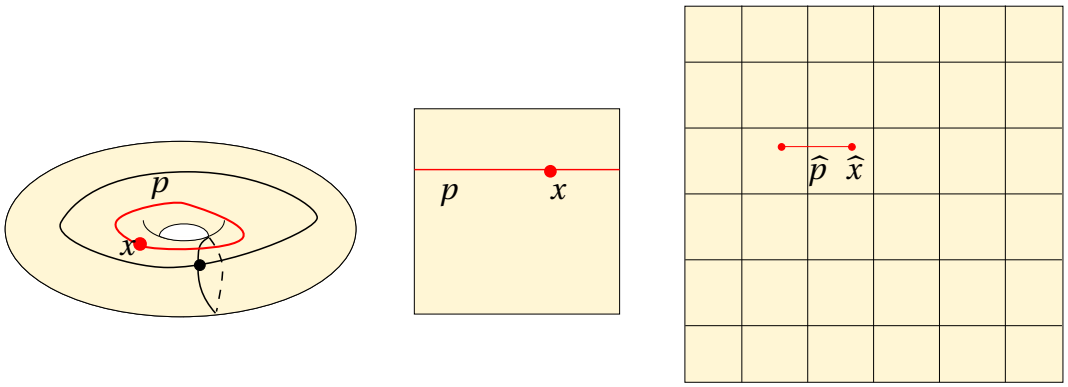


Figure 3.18: Lifting a loop p on the torus to a path on its universal cover.

definition of covering spaces: when one sits at a point $\widehat{x}$ of the covering space, the local homeomorphism π specifies how to move on $\widehat{S}$ so that one follows the path p.

Every surface has a unique covering space $\widetilde{S}$ that is simply connected, that is, where every loop in $\widetilde{S}$ is homotopic to a trivial loop, it is called the **universal cover** of S. If S is a sphere, its universal cover is itself, so let us assume that it is not the case. One way to build this cover is as follows: pick a graph G cellularly embedded on S with a single vertex and a single face (for example one of the graphs used in the classification theorem). Cutting S along G gives a polygon and one can **tile** the plane with this polygon by putting adjacent copies of this polygon next to each other, so that every vertex of the tiling is adjacent to the correct number of polygons. This construction is pictured in Figure 8.20 for the torus.

For surfaces of higher genus, the same construction works, but the tiling will not look as symmetric : indeed it is for example impossible to tile the plane with regular octagons. This is not an issue for our construction, since any tiling with octagons will do, even if they do not have the same shape. However, an insightful way to deal with this issue is to use **hyperbolic geometry**: it is a non-Euclidean geometry on the open disk that allows for regular tilings of polygons with an arbitrary number of faces.

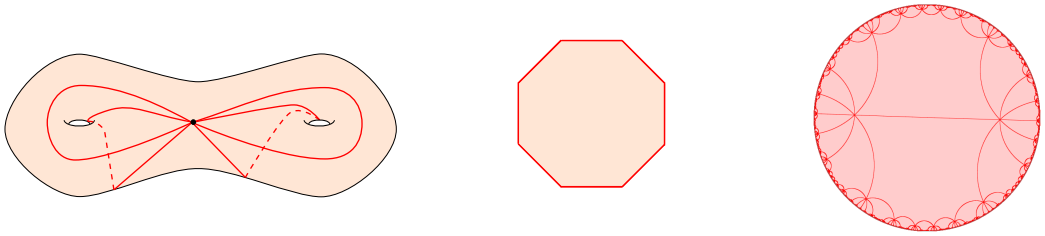


Figure 3.19: The universal covering space of the genus 2 surface.

Figure 3.19 pictures the universal cover of a genus 2 surface as a hyperbolic tiling.

One can readily check that the spaces we obtain are universal covering spaces of their respective surfaces, since they are simply connected and the map π can naturally be inferred by the tiling. One key property of universal covers is that a loop on S is contractible if and only if all its lifts in the universal cover are also loops, as can be tested on the above examples. This will be leveraged in the next section to design an algorithm to test contractibility of loops on surfaces.

Remark: The irruption of hyperbolic geometry here is not random at all: one can show that surfaces of genus at least 2 do not admit Euclidean metrics, but do admit hyperbolic ones. It is the lift of such a metric that one uses to obtain a hyperbolic tiling. Hyperbolic geometry plays a primordial role in the study of the geometric properties of surfaces, and has been used increasingly as well in the design of algorithms for computational topology.

4

The Homotopy Test

Contents

A fundamental problem when dealing with curves on surfaces is to decide if a given closed curve can be contracted to a point, or more precisely to a constant curve. This is sometimes referred to as the **contractibility problem**. More generally, we can ask whether two closed curves on a surface are related by a continuous deformation. This question has two variants: we may or may not require the curves to share a given point that remains fixed during the deformation. Note that the problem with fixed point has an obvious reduction to the contractibility problem. Indeed, two curves c, d are homotopic with fixed point if and only if the concatenation $c \cdot d^{-1}$ is contractible. Without the fixed point requirement, that is when the curves are allowed to move *freely* on the surface, the problem is known as the **transformation problem** and can be expressed as a **conjugacy problem**. To see this, choose a point v on a surface S

and suppose that c and d are homotopic[1]. We can deform c and d so that each of them passes through p. The resulting curves are still homotopic. In other words, there is a continuous mapping $h : \mathbb{S}^1 \times [0,1] \to S$ such that $h_{|\mathbb{S}^1 \times \{0\}} = c$ and $h_{|\mathbb{S}^1 \times \{1\}} = d$, and viewing $\mathbb{S}^1 \times [0,1]$ as an annulus, each boundary has a point sent on v by h. We connect these two points by a simple path a in the annulus. The map h sends this path to a closed path α. See Figure 4.1. Cutting the annulus through a we obtain a disk whose

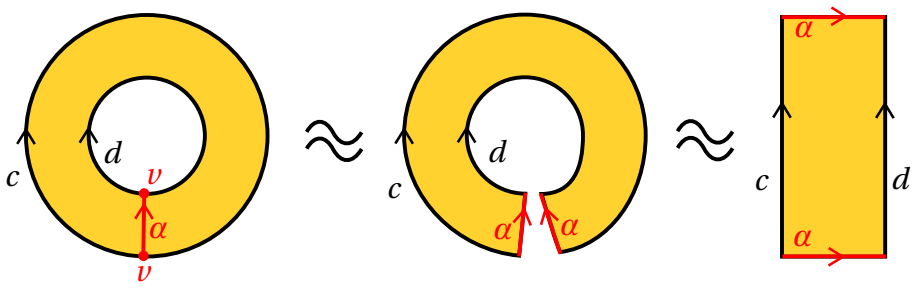


Figure 4.1: c and d are homotopic if and only if their homotopy classes are conjugate.

boundary is sent to $c \cdot \alpha \cdot d^{-1} \cdot \alpha^{-1}$ which is thus contractible. Hence, c is homotopic to $\alpha \cdot d \cdot \alpha^{-1}$ or, equivalently, the homotopy classes of c and d are conjugate in the fundamental group $\pi_1(S, v)$. For the reverse implication, if c and d have conjugate homotopy classes we can just read Figure 4.1 from right to left and conclude that c and d are indeed homotopic.

4.1 Dehn's Algorithm

Suppose that S is a **reduced** combinatorial surface, that is a map with a single vertex and a single face. Its graph G is thus composed of loop edges, each of which corresponds to a generator of the fundamental group of S. We can directly read the homotopy class of a closed path in G: the sequence of arcs of the path translates to the product of the corresponding generators and their inverses. This product is often viewed as a **word** on the generators and their inverses, so that the contractibility problem is the same as the **word problem** where we ask if a product of generators and their inverses is the trivial element in the fundamental group of S.

Max Dehn was among the first to establish and exploit the connection between Topology (the contractibility problem) and Algebra (the word problem). He proposed a solution to the word problem now known as **Dehn's algorithm** [Sti87, paper 5]. Dehn observed that the lift of G in the universal covering space of S induces a tessellation of the plane composed of copies of the unique polygonal face of G in S. This tessellation is actually the **Cayley complex** of $\pi_1(S, v)$ where v is the unique vertex of G. This complex $\tilde{S}$ is relative to the set of generators $\{\beta_i\}_i$ of $\pi_1(S, v)$ – the homotopy classes of the loop edges in G – and to their relation F obtained from the unique facial walk of G in S. The vertex set of $\tilde{S}$ are the elements of $\pi_1(S, v)$ and there is an (oriented) edge labelled β_i between every $\alpha \in \pi_1(S, v)$ and $\alpha \cdot \beta_i$. Finally, disks are glued along each

[1] Homotopy without fixed point is often called *free* homotopy. For concision, we drop the term free. In general, it should be clear from the context whether we use free homotopy or homotopy with fixed point, and we will specify when necessary that the homotopy is with fixed point.

closed path labelled by F in the resulting graph. If a closed path c in G is contractible in S, then any of its lifts is a closed path in $\tilde{S}$. Dehn further claims that
any closed path in $\tilde{S}$ contains either a ***spur****, i.e. an arc followed by its opposite arc, or more than half of F, i.e. a subpath labelled by some word U such that for some other V shorter than U, the concatenation UV is a cyclic permutation of F or its inverse.*
In both cases c is homotopic to a shorter closed path obtained by removing the spur in the former case and by replacing the path labelled by U with the complementary path labelled by V^{-1} in the latter case. This leads to an algorithm where we inductively search for spurs or large pieces of F until we obtain a word that we cannot reduce anymore. It then follows from Dehn's claim that c is contractible if and only if this word is empty.

In order to prove his claim, Dehn notes that the faces of the complex $\tilde{S}$ are arranged in rings of faces $R_1, R_2, \ldots$, where R_1 is the set of faces incident with a given vertex[2] v_0 of $\tilde{S}$ and R_{i+1} is the set of faces not in R_i sharing a vertex with the external boundary of R_i. Remark that a face of R_{i+1} has at most two vertices in R_i. Hence, if S is an orientable surface of genus $g \geq 2$, each face has $4g$ sides and a face of a ring has at least $4g-2 > 2g$ vertices on its external boundary. Consider now a closed path $\tilde{c}$ without spurs and passing through v_0. Let i be maximal such that $\tilde{c}$ contains a vertex of the external boundary of R_i. Figure 4.2 illustrates a factious case of a relation of length 6. Since $\tilde{c}$

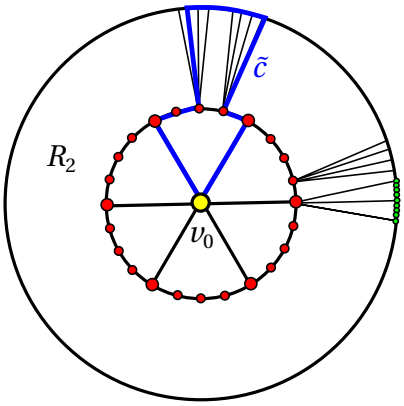


Figure 4.2: The faces of the complex $\tilde{S}$ are arranged in rings of faces.

has no spurs it is easily seen that it contains the whole intersection of a face with the external boundary of R_i. The previous remark allows to conclude the claim.

Dehn's algorithm has a simple implementation that runs in $O(g|c|)$ time where g is the genus of S. A more careful implementation with $O(g+|c|\log g)$ time complexity was described by Dey and Schipper [DS95]. Finally, optimal $O(g+|c|)$ algorithms were proposed [LR12, EW13]. We shall describe these last approaches to the contractibility and deformation problems, not so far from Dehn's original approach but including more recent techniques borrowed from geometric group theory.

[2] In his original work, Dehn defines R_1 as a single face.

4.2 van Kampen Diagrams

4.2.1 Disk Diagrams

A useful tool concerning contractible curves is provided by the so called **van Kampen diagrams**. Such diagrams bear different names in the litterature, among which **disk diagrams** and **Dehn diagrams** are the most common. Intuitively, a disk diagram allows to express the combinatorial counterpart of the following characterization of contractible loops in a topological space X: a loop $\mathbb{S}^1 = \partial\mathbb{D}^2 \to X$ is contractible if and only if it extends to a continuous map $\mathbb{D}^2 \to X$, where $\mathbb{D}^2$ is the unit disk. Given a combinatorial map M with graph G, a **disk diagram over** M is a combinatorial sphere D with a marked **outer face**, and a labelling of the arcs of D by the arcs of M such that opposite arcs are labelled by opposite arcs and such that every facial walk of D that is not the outer face is labelled by some facial walk of M. In other words, D is a gluing of faces and edges of M that is homeomorphic to the complement of an open disk in a sphere. For instance, this complement could be a tree. In general, it is a tree-like arrangement of topological closed disks connected by trees. The facial walk of the outer face of D is denoted by ∂D. The diagram is **reduced** if any two of its *inner* faces (*i.e.* not the outer one) sharing a vertex v are labelled by facial walks that are not inverse to each other when starting the facial walks at v.

Lemma 4.2.1 (van Kampen, 1933)**.** *A closed path c in M is contractible if and only if it is the label of the outer facial walk of a reduced disk diagram over M.*

The proof uses the intuitive fact that homotopic closed paths are **combinatorially homotopic**, where a combinatorial homotopy is a sequence of **elementary homotopies** that consist in either inserting or removing a spur, or replacing a subpath of a facial walk by the complementary subpath. See Theorem 4.7 in the previous lecture notes.

PROOF OF LEMMA 4.2.1. We first prove the existence of a not necessarily reduced disk diagram. Let $c_0 = 1 \to c_1 \to \cdots \to c_k = c$ be a sequence of k elementary homotopies attesting the contractibility of c, where 1 denotes a constant path. By induction on k, we may assume the existence of a disk diagram D such that ∂D is labelled by c_{k-1}. There are three cases to consider.

- If $c_{k-1} \to c_k$ consists in inserting a spur aa^{-1}, then we can form a disk diagram for c_k by attaching a pendant edge labelled with a to the boundary of D.
- If $c_{k-1} \to c_k$ consists in removing a spur, then either this spur corresponds to two consecutive arcs of ∂D with distinct edge support or it corresponds to the two arcs of a single pendant edge. In the former case, we form a disk diagram for c_k by gluing the two arcs along ∂D. In the latter case, we contract the pendant edge.
- Otherwise, $c_{k-1} \to c_k$ consists in the replacement of a subpath p by a subpath q such that pq^{-1} is a facial walk of M. We then perform a subdivision of the outer face of D, inserting a new edge between the extremities of p. The new

outer face is chosen among the two new faces as the one not bounded by p. We next subdivide the new edge $k-1$ times, where k is the number of arcs of q. We finally extend the labelling trivially by sending the subdivided edge to the edges of q. This amounts to glue a face with facial walk pq^{-1} along p on D.

If the resulting diagram is not reduced, then there are two facial walks sharing a vertex v and labelled by opposite facial walks of M. We "open" D at v and identify the two facial walks according to the labels of their arcs. This produces a new diagram with two faces less and does not modify the outer face boundary. We repeat the procedure as long as the diagram is not reduced. By induction on the number of faces this procedure must end. Note that the final diagram may have no face, in which case its graph must be a tree corresponding to a closed path that can be reduced to a point by removing spurs only. □

Exercise 4.2.2. Relates the degree of an inner vertex in a reduced disk diagram over M with the degree of the corresponding vertex in M.

4.2.2 Annular Diagrams

There is an analogous notion of **annular diagram** defined by a combinatorial sphere with two marked outer faces instead of one.

Lemma 4.2.3 (Schupp, 1968)**.** *Two closed paths c and d in M are homotopic if and only if there exists a reduced annular diagram over M such that the facial walks of its outer faces (oriented consistently) are labelled with c and d respectively.*

PROOF. By the introductory discussion there exists a path p such that $c \cdot p \cdot d^{-1} \cdot p^{-1}$ is contractible. By Lemma 4.2.1, there exists a disk diagram over M whose boundary is labelled with $c \cdot p \cdot d^{-1} \cdot p^{-1}$. We may identify the subpaths corresponding to p and p^{-1} respectively and get an annular diagram whose perforated faces are labelled with c and d. If the diagram is not reduced, we proceed as in the proof of Lemma 4.2.1. □

4.3 Gauss-Bonnet Formula

Another interesting tool is given by a combinatorial version of the famous Gauss-Bonnet theorem. This theorem relates the curvature of a Riemannian surface S (say a smooth surface embedded into $\mathbb{R}^3$) with its Euler characteristic χ, hence a local geometric quantity with a global topological one. If K is the Gauss curvature of S and k_g is the geodesic curvature along its (smooth) boundary ∂S then:

$$\int_S K \, \mathrm{d}s + \int_{\partial S} k_g \, \mathrm{d}\ell = 2\pi\chi \tag{4.1}$$

We can obtain a combinatorial version of this formula using some kind of angle structure over a combinatorial surface. Given an orientable combinatorial map $M = (A, \rho, \iota)$, we consider an angular assignment of its corners, that is a real function θ defined

over the set of corners. Here, a **corner** is any pair $(a, \rho(a))$, for $a \in A$, of successive arcs around a vertex. We require that the sum of the angular assignments of the corners of any face f satisfies

$$\sum_{c \in f} \theta(c) = d_f/2 - 1, \tag{4.2}$$

where d_f is the degree of the face, *i.e.* the length of its facial walk. Intuitively, this condition amounts to assume that the faces are Euclidean polygons if we view an angular assignment as a normalized angle, measuring angles in terms of parts of a circle instead of radians. Indeed, the total angle of a Euclidean polygon with d_f sides is $(d_f - 2)\pi$, which is $d_f/2 - 1$ when normalized. We then define the curvature of an interior vertex v as

$$\kappa(v) = 1 - \sum_{c \in v} \theta(c), \tag{4.3}$$

where, $c \in v$ indicates that the corner $c = (a, \rho(a))$ is incident to the source vertex v of a. We also define the (geodesic) curvature of a boundary vertex[3] v as

$$\tau(v) = 1/2 - \sum_{c \in v} \theta(c) \tag{4.4}$$

Those curvatures thus measure the angle default with respect to the flat situation ($\kappa = 1$ and $\tau = 1/2$). They can be related to the Gauss curvature of the flat conic surface S_v with one singularity at v obtained by gluing small isocele triangles, one for each corner $c \in v$, with angle $2\pi\theta(c)$ at v. The boundary of S_v is a broken line so that Formula (4.1) should be corrected with the term $\sum_w (\pi - \alpha_w)$, where w runs over the boundary vertices of S_v and α_w is the interior angle at w. Since the geodesic curvature of a line segment is zero, Formula (4.1) becomes

$$\int_{S_v} K \, \mathrm{d}s + \sum_w (\pi - \alpha_w) = 2\pi\chi = 2\pi$$

Noting that with $\sum_w (\pi - \alpha_w)$ is the sum of the angles at the corners of v we obtain $\int_{S_v} K = 2\pi\kappa_v$.

Theorem 4.3.1 (Combinatorial Gauss-Bonnet —)**.** *Let M be a combinatorial map whose boundary is composed of disjoint simple cycles in the graph of M. Denote by χ the Euler characteristic of M and by $V^o \cup V^\partial = V$ its interior and boundary vertex sets. Then, for any angular assignment, we have*

$$\sum_{v \in V^o} \kappa(v) + \sum_{v \in V^\partial} \tau(v) = \chi$$

It is possible to drop the condition on the boundary of M using a slightly different notion of curvature, see Erickson and Whittlesey [EW13]. The present presentation is inspired by Gersten and Short [GS90] and makes the parallel with the differentiable version rather transparent.

[3] Formally, a combinatorial surface with boundary is defined by marking some faces as perforated, and a boundary vertex is any vertex incident to a perforated face.

PROOF. By definition, we compute

$$\sum_{v\in V^o}\kappa(v)=|V^o|-\sum_{c\in v\in V^0}\theta(c)\quad\text{and}\quad\sum_{v\in V^\partial}\tau(v)=|V^\partial|/2-\sum_{c\in v\in V^\partial}\theta(c)$$

It follows that $\sum_{v\in V^o}\kappa(v)+\sum_{v\in V^\partial}\tau(v)=|V|-|V^\partial|/2-\sum_{c\in v\in V}\theta(c)$. By distributing the corners according to faces rather than vertices and by the angular assignment requirement (4.2), we see that

$$\sum_{c\in v\in V}\theta(c)=\sum_{c\in f\in F}\theta(c)=\sum_{f\in F}(\frac{d_f}{2}-1)=\frac{1}{2}\sum_{f\in F}d_f-|F|$$

where F is the set of faces of M. Since every arc appears in exactly one facial walk, except for those on the boundary of M, we have: $\sum_{f\in F}d_f=2|E|-|E^\partial|$ where E and E^∂ are the set of edges and boundary edges respectively. Since $|E^\partial|=|V^\partial|$, we conclude that

$$\begin{aligned}\sum_{v\in V^o}\kappa(v)+\sum_{v\in V^\partial}\tau(v) &= |V|-|V^\partial|/2-(|E|-|E^\partial|/2)-|F|)\\ &= |V|-|E|+|F|\end{aligned}$$

□

4.4 Quad Systems

From an algorithmic point of view it is more convenient to work with combinatorial surfaces all of whose faces are quadrilaterals. We call such a surface a **quadrangulation** or a **quad system**. Given a combinatorial surface without boundary, we easily get a quadrangulation of the same topological surface as follows. We insert a vertex inside each face and connect this vertex to all the corners of the face. Hence, if a facial walk has length k we introduce k new edges in the face. This subdivides each face into triangles. We then delete all the edges of the original graph, thus merging all the triangles by pairs to form quadrilaterals. In practice, we will also require that the vertices have a high degree, say at least 8. For a surface of genus $g\geq 2$ this is easily obtained by first reducing the combinatorial surface to a single vertex and a single face before applying the above quadrangulation process. The resulting quadrangulation has two vertices, $4g$ edges and $2g$ quadrilaterals. Figure 4.3 shows a reduced surface and its quadrangulation.

Lemma 4.4.1. *Let Q be a quadrangulation derived by the previous process from a given map M without boundary. We can preprocess M in linear time (proportional to its number of arcs) so that any closed walk c can be transformed in $O(|c|)$ time into a homotopic closed walk of size at most $2|c|$ in Q.*

To see this, consider a spanning tree T of the graph G of M. Contracting T gives a surface M' with graph G/T and with a single vertex. Next consider a spanning tree of the dual graph of M' and denote by L the corresponding set of primal edges. Deleting

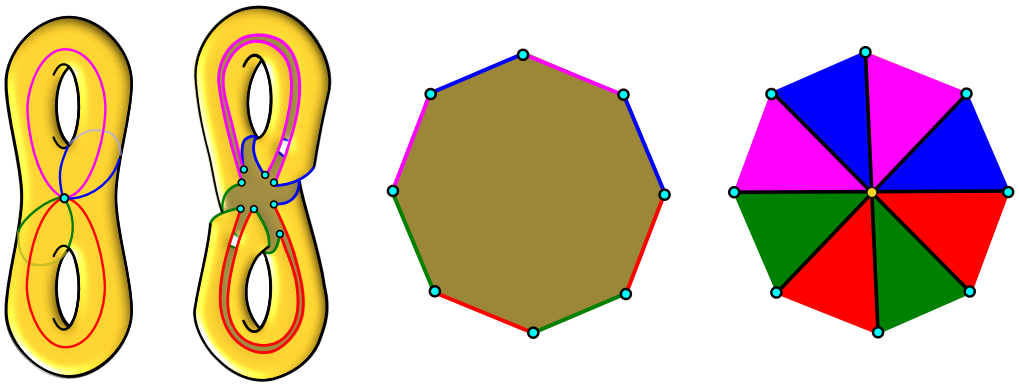


Figure 4.3: From left to right, a reduced surface is cut-opened and its unique face is triangulated by inserting a vertex in the center. Triangles of the same color are merged by deleting the original loop edges.

the edges in L leaves a reduced surface M'' and we construct Q by first inserting a new vertex z in the unique face of M'' together with all the edges from z to the corners of the face. We finally remove the remaining edges of G/T to get Q. Note that any edge e of G/T is homotopic to the path of length two in Q connecting z to the two endpoints of e. We can precompute and store these length two paths for each e in total linear time. Now, given any c, we contract all the occurrences of edges of T in c to obtain a homotopic closed walk c' in M'. We further replace every remaining edge by the corresponding length two path to obtain a homotopic closed walk as desired in Q. This transformation takes $O(|c|)$ time.

Exercise 4.4.2. Propose a construction of quadrangulation starting from a combinatorial surface with nonempty boundary. Can you extend Lemma 4.4.1 accordingly?

4.5 Canonical Representatives

The last and most important ingredient of the homotopy test is the construction of a canonical representative in each free homotopy class. Given a closed walk in a quadrangulation, the idea is to shorten the walk as much as possible to obtain a combinatorial geodesic. As a homotopy class may contain several geodesics, we further consider the *rightmost* geodesic to define a canonical representative. Once a canonical representative has been computed for two given closed walks we can decide if the walks are homotopic by just checking if their representative are equal up to a circular permutation. The shortening process is based on successive simplifications of spurs and brackets as explained below.

4.5.1 The Four Bracket Lemma

Let (a_1, a_2) be a pair of arcs sharing their origin vertex v on a quadrangulation M. Following the terminology of Erickson and Whittlesey [EW13], we define the **turn** of (a_1, a_2) as the number of corners between a_1 and a_2 in counterclockwise order around

v. Hence, if v is a vertex of degree d in M, the turn of (a_1, a_2) is an integer modulo d that is zero when $a_1 = a_2$. The **turn sequence** of a subpath $(a_i, a_{i+1}, \ldots, a_{i+j-1})$ of a closed walk of length ℓ is the sequence of $j+1$ turns of $(a_{i+k}^{-1}, a_{i+k+1})$ for $-1 \le k < j$, where indices are taken modulo ℓ. The subpath may have length ℓ, thus leading to a sequence of $\ell+1$ turns. Note that the turn of $(a_{i+k}^{-1}, a_{i+k+1})$ is zero precisely when (a_{i+k}, a_{i+k+1}) is a spur. A **bracket** is any subpath whose turn sequence has the form 12^*1 or $\bar{1}\bar{2}^*\bar{1}$ where t^* stands for a possibly empty sequence of turns t and $\bar{x}$ stands for $-x$. Intuitively, if we imagine that every corner of M has a right angle, a bracket corresponds to a straight path ending with right angles. A quadrangulated disk is **non-singular** if its boundary is a simple cycle of its graph.

Lemma 4.5.1 (Four bracket —, [GS90, EW13])**.** *Let D be a non-singular quadrangulated disk all of whose interior vertices have degree at least four. Then, the boundary of D contains at least four brackets.*

Figure 4.4 illustrates the Lemma.

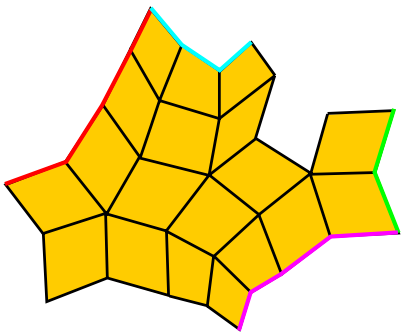


Figure 4.4: The quadrangulated disk has four highlighted brackets. Can you find them all?

PROOF. Consider the constant angular assignment $1/4$ over D. By the Gauss-Bonnet theorem 4.3.1, we have $\sum_{v \in int D} \kappa(v) + \sum_{v \in \partial D} \tau(v) = \chi(D) = 1$. By (4.3), every interior vertex has non-positive curvature. It follows that

$$\sum_{v \in \partial D} \tau(v) \ge 1 \tag{4.5}$$

Remark that $\tau(v) = (2 - c_v)/4$ where c_v is the number of corners incident to the boundary vertex v. Call v **convex, flat** or **concave** if $c_v = 1$, $c_v = 2$ or $c_v \ge 3$ respectively. In other words v is convex, flat or concave if its curvature is respectively $1/4$, zero or negative. Inequality (4.5) implies that the boundary of D contains at least four more convex vertices than concave vertices. The lemma easily follows. □

Corollary 4.5.2. *A nontrivial contractible closed walk in a quadrangulation all of whose interior vertices have degree at least four contains either a spur or a bracket.*

PROOF. Suppose that a nontrivial contractible closed walk c has no spurs. By the van Kampen Lemma 4.2.1, c is the label of the boundary of a reduced disk diagram D. Let H be the dual graph of D: it has one dual vertex per quadrilateral of D and one dual edge for each pair of quadrilaterals sharing an edge. If H is connected then D is non-singular. Indeed, if the boundary of its outer facial walk ∂D was not a cycle it would contain a degree one vertex, which would contradicts that c has no spurs. We can thus apply the four brackets Lemma 4.5.1 to conclude that ∂D has at least one bracket. However, the turn t at a vertex of ∂D is the same as the turn of the corresponding vertex in c (up to a multiple of the degree of that vertex in the quadrangulation). It follows that c has also a bracket. If H is not connected, then D consists of a tree-like arrangement of non-singular disks connected by trees through cut vertices. This arrangement has a "degree one" non-singular disk connected to the rest through a single cut vertex. By the four vertex theorem this disk has four brackets, two of which do not contain the cut vertex. These two brackets thus correspond to brackets in c. □

Exercise 4.5.3. Show that we can actually claim the existence of a spur or *four* brackets in Corollary 4.5.2.

4.5.2 Bracket Flattening

A **bracket flattening** consists in replacing a bracket and the two incident edges with the "straight line" between their endpoints. Some care must be taken when the incident edges of the bracket share their endpoints or when these edges are part of the bracket. Figure 4.5 depicts the different cases. Corollary 4.5.2 provides a practical algorithm to

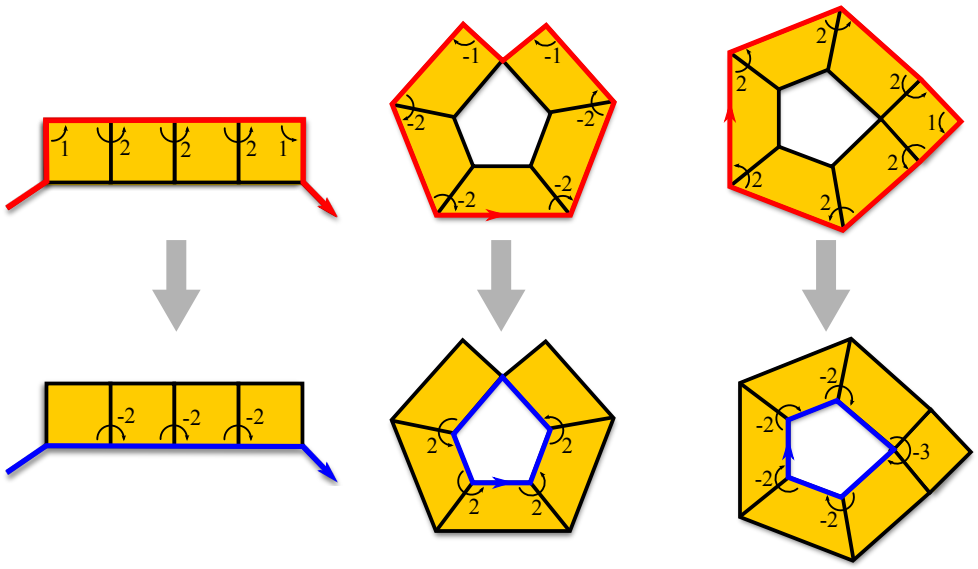


Figure 4.5: Left, a typical bracket flattening. Middle, the edges incident to the bracket share their endpoints. Right, the bracket covers the whole closed walk.

test if a given closed walk c is contractible: remove the spurs and flatten the brackets until there is no more. Then c is contractible if and only if the resulting walk is reduced

to a vertex. Since each spur removal or bracket flattening decreases the number of edges by two the number of steps is linear in $|c|$. Note that the non-typical bracket flattening (Figure 4.5, Right) may only occur when c is non-contractible (why?).

4.5.3 Canonical Representatives

A homotopy class may contain distinct closed walks without spurs and brackets. In order to get a canonical representative in each homotopy class we further push such reduced walks as much as possible "to their right". Say that a vertex of a walk is **convex** if its turn is 1 in the turn sequence of the walk. If a closed walk c contains a convex vertex v we consider the maximal subpath including v whose turning sequence has the form $x2^*12^*y$, where $x, y \neq 2$. This subpath, say p, bounds an L-shaped sequence of quadrilaterals that lies to its right. Replacing p by the complementary path bounding the sequence of quadrilaterals gives a closed walk homotopic to c with one less convex vertex. Note that this replacement does neither introduce a bracket nor a spur. Some care must again be taken when p covers c. See Figure 4.6 for all the possible typical and non-typical configurations. A right push reduces the number of convex vertices

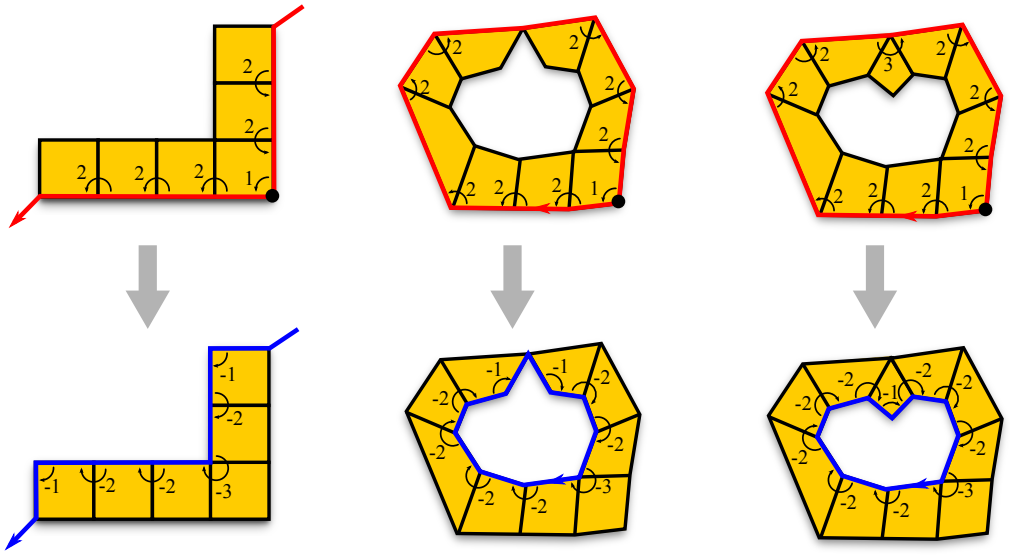


Figure 4.6: The different configurations for a right push.

by one, so that only a linear number of pushes can be applied. A last exceptional case occurs when the turn sequence of c is composed of 2's only. We also apply a right push in this case, which transforms the turn sequence into a sequence of $\bar{2}$ as on Figure 4.7. When no right pushes apply, the closed walk is said **reduced** .

Proposition 4.5.4. *Let M be a quadrangulation all of whose vertices have degree at least five. Then each homotopy class contains a unique reduced closed walk.*

PROOF. Let c and d be homotopic reduced closed walks. We need to show that $c = d$. Following Lemma 4.2.3 we consider a reduced annular diagram A for c and d. We first claim that the two boundaries of A are simple. Otherwise, one boundary has a

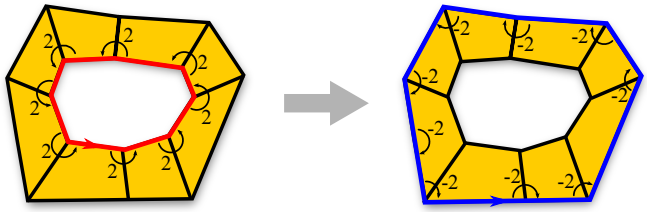


Figure 4.7: In case all the turns are equal to 2, we push the walk to the right to obtain a sequence $\bar{2}^*$ of turns.

cut vertex that separates A into a smaller annular part A' and a disk part D connected to A' through a single cut vertex. By the four brackets theorem, the boundary of D has one (in fact at least two) bracket disjoint from this cut vertex. In turn, this bracket would appear in c or d, contradicting the hypothesis that c and d are reduced.

- If the two boundaries of A have a common vertex then cutting through that vertex gives a disk diagram D' bounded by (circular permutations of) c and d. This diagram is a tree-like arrangement of non-singular disks connected by trees through cut vertices. For convenience, we also call cut vertices the two common endpoints of c and d. If a non-singular disk is incident to a single cut vertex, then it is bounded by a subpath of one of c or d. By the four bracket theorem this subpath would contain a bracket, in contradiction with the reduction hypothesis. It follows that D' is a linear sequence of non-singular disks connected by simple paths (otherwise c or d would have a spur). We claim that none of those non-singular disks can have an interior vertex. Otherwise, considering the constant angular assignment $1/4$ over D', this interior vertex would have negative curvature. An argument similar to the proof of the four bracket theorem 4.5.1 shows that the boundary of D' would contain five brackets, one of which not incident to any cut vertex. This would again lead to the contradiction that c or d has a bracket. The dual graph of each non-singular disk is thus a tree. However, no matter the shape of this tree and no matter how its boundary is split one of the resulting boundary paths would contain a bracket or a convex vertex. In both cases this would contradict the fact that c and d are reduced. It follows that D' has no non-singular disk, hence is a simple path, implying that $c = d$.

- Suppose now by way of contradiction that the two boundaries of A are disjoint. Then we can argue similarly as above that A has no interior vertex. The dual graph of A is thus a single cycle with some attached trees. It must actually be a cycle, since otherwise one of the boundaries of A would have a bracket. This cycle has to go straight without bending since otherwise c or d would have a convex vertex or a bracket. (This last case occurs even with a single bend as on Figure 4.5, Right.) It follows that one of the boundaries of A has 2-turns only as on right Figure 4.7, contradicting that c and d are reduced. In any case we have reached a contradiction, so that the boundaries of A cannot be disjoint.

□

A reduced closed walk can thus play the role of **canonical representative** for its homotopy class.

4.6 The Homotopy Test

We now have all the necessary ingredients to perform a linear time homotopy test. Thanks to Lemma 4.4.1, we can assume given two closed walks in a quadrangulation. We compute the canonical form of each closed walk by first removing spurs and brackets as described in Sections 4.5.2. We can first remove all the spurs in linear time. The flattening of a bracket may introduce new spurs but their removal can be charged to the removed edges, so that the total time spent to remove spurs is still linear in the end. Note that a flattening transforms a bracket into a flat part (a run of 2-turns or of $\bar{2}$-turns) that may be part of a larger flat part. In order to avoid loosing time for traversing several times the same flat parts, we add jump pointers between the endpoints of each flat part before we perform any flattening. We also store the turns at these endpoints and the length of the flat part. Then, after each bracket flattening we update the turns at the endpoints and check if the resulting flat part should be merged with the at most two surrounding flat parts. This can be done in constant time thanks to the jump pointers. This way each bracket flattening costs a constant time. Since each flattening decreases the number of edges, there can only be a linear number of them and the total cost for removing spurs and brackets is thus linear. The sequence of edges of the resulting closed walk is easily recovered from the jump pointers and the lengths of the flat parts. We just need to know one edge along the walk, which we can update easily as spurs and brackets are simplified.

Once spurs and brackets have been removed we obtain a geodesic that needs to be pushed to its right as described in Section 4.5.3. Each right push transforms a subpath of the geodesic into another subpath of the same length without 1-turns or 2-turns. It follows that none of the vertices of this subpath will be pushed again. The total time needed to obtain a rightmost geodesic is thus linear. This shows that

Theorem 4.6.1. *The canonical representative of a closed walk c in a quadrangulation, all of whose vertices have degree at least five, can be computed in $O(|c|)$ time.*

Corollary 4.6.2. *Given two closed walk of length at most ℓ in a combinatorial map of size n we can decide if they are homotopic in $O(n+\ell)$ time.*

PROOF. According to Lemma 4.4.1, we can reduce the combinatorial map to a quandrangulation in $O(n)$ time and get closed walks homotopic to the given one in $O(\ell)$ time. By Theorem 4.6.1 we can compute the canonical form of the walks in $O(\ell)$ time. Now these canonical forms, say c and d, are homotopic if and only if one is a circular permutation of the other. This can be tested in linear time by checking whether c is a substring of $d \cdot d$ thanks to the Knuth-Morris-Pratt string searching algorithm [KMP77] [CLRS02, Sec. 32.4]. □

5

Minimum Weight Bases

Contents

In the second lecture we saw that a graph could be associated with a vector space, called the cycle space. We will see that this cycle space can be extended to surfaces giving birth to the *first homology group*. We also introduced the fundamental group of a graph or of a surface in another lecture. Hence, we now have two group structures that encode the topology of a space X, where X is either a graph or a surface. These structures are both generated by closed walks in the graph of X and we call a **basis** any generating set with the minimum number of closed walks. In order to derive a more informative notion of minimality we assume that the edges of the considered graph have a positive weight. This allows to define the weight of a closed walk as the sum of its edge weights (counted with multiplicity). A **minimum weight basis** is then a

basis such that the sum of the weights of its members is minimum. The computation of minimum weight bases has received much attention when X is a graph and was studied more recently for combinatorial surfaces. Good references on the subject include a comprehensive survey on cycle bases in graphs by Kavitha et al. [KLM+09] and another survey on optimization of cycles and bases on surfaces by Erickson [Eri12]. We shall use the qualifiers *minimum* and *shortest* interchangeably to designate a walk, tree or subgraph of minimum weight.

5.1 Minimum Basis of the Fundamental Group of a Graph

Let G be a connected graph with basepoint v and let $|.| : E \to \mathbb{R}_+$ be a weight function. The fundamental group $\pi_1(G, v)$ is a free group whose rank is the number of chords of any spanning tree of G, which is $1 - n + m$, where n and m are respectively the number of vertices and edges of G. Indeed, as we saw, every chord e of a spanning tree T gives rise to a loop $\gamma^T_{v,e}$ obtained by connecting v to each endpoint of the chord using paths in the tree, and these loops form a basis of $\pi_1(G, v)$. Not all bases arise this way but a minimum one may indeed be obtained by this construction. For this, we take for T a **shortest path tree** with root v: every vertex w of G is linked to v by a path in T whose weight is minimum among all the paths from v to w in G. When all the weights are equal a shortest path tree can be computed by a breadth-first search traversal in time $O(m)$. In the general case, one may use Dijkstra's algorithm [CLRS09] to compute a shortest path tree in $O(m + n \log n)$ time. Remark that $\gamma^T_{v,e}$ is a shortest loop through the chord e.

Theorem 5.1.1. *The basis of $\pi_1(G, v)$ associated with a shortest path tree with root v is a minimum weight basis.*

The following proof is based on an purely algebraic preliminary lemma. First note that a free group F over a set $(x_1, x_2, \ldots, x_r)$ gives rise to a *free Abelian group* (this is the same a free $\mathbb{Z}$-module) F^{ab} by making all the generators commute. Hence, if we let R be the set of relations $\{x_i x_j = x_j x_i\}_{1 \le i < j \le r}$, a presentation for F^{ab} is $< \{x_1, x_2, \ldots, x_r\} \mid R >$. We thus have a quotient $F \twoheadrightarrow F^{ab} = F/ < R >$ and we denote by $[x] \in F^{ab}$ the image of any $x \in F$. Note that $[x]$ can be uniquely written as a linear combination of the $[x_i]$'s.

Lemma 5.1.2. *Let $(x_1, x_2, \ldots, x_r)$ and $(y_1, y_2, \ldots, y_r)$ be two bases of a free group F. Denote by $y_j(x_1, x_2, \ldots, x_r)$ the expression of y_j in terms of the basis $(x_1, x_2, \ldots, x_r)$. Then, there exists a permutation σ of $\{1, \ldots, r\}$ such that for each i the coefficient of $[x_i]$ in $[y_{\sigma(i)}(x_1, x_2, \ldots, x_r)]$ is nonzero.*

PROOF. The automorphism of F defined by $x_i \mapsto y_i$, $1 \le i \le r$, quotients to an automorphism of F^{ab}. Let c_{ij} be the coefficient of $[x_j]$ in $[y_i(x_1, x_2, \ldots, x_r)]$. Viewing F^{ab} as a free $\mathbb{Z}$-module over the $[x_i]$'s, the matrix $(c_{ij})_{1 \le i,j \le r}$ of this automorphism has nonzero determinant. It follows that at least one term $\prod_{1 \le i \le r} c_{i\sigma(i)}$ of the usual Leibnitz expansion of the determinant must be nonzero. This implies the lemma. □

PROOF OF THEOREM 5.1.1. Let T be a shortest path tree from v. We denote by $e_1, e_2, \ldots, e_r$ the chords of T in G. Let $(b_1, b_2, \ldots, b_r)$ be a basis for $\pi_1(G, v)$. According to the preliminary lemma, there is a permutation σ of $\{1, \ldots, r\}$ such that the coefficient of $[\gamma^T_{v,e_i}]$ in $[b_{\sigma(i)}]$ is nonzero. It follows that $b_{\sigma(i)}$ goes through e_i, hence is at least as long as γ^T_{v,e_i} by the remark before the theorem. As a direct consequence $\sum_i |b_i| \geq \sum_i |\gamma^T_{v,e_i}|$. □

5.2 Minimum Basis of the Cycle Space of a Graph

As we saw, the set of Eulerian subgraphs $Z(G)$ of a connected graph G can be given a vector space structure over the coefficient field $\mathbb{Z}/2\mathbb{Z}$. We also observed that a basis could be obtained from any spanning tree T of G by considering for each chord e of T the cycle γ^T_e composed of e and the path in T connecting e's endpoints. Such a basis is called a *fundamental cycle basis*. As opposed to the case of the fundamental group, a minimum weight basis of the cycle space is not always a fundamental cycle basis. The counterexample in Figure 5.1 was found by Hartvigsen and Mardon [HM93]. In general, looking for the minimum weight *fundamental* basis is NP-hard [DPeK82].

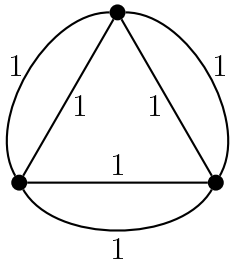


Figure 5.1: Each spanning tree in this graph is a path of length 2. The corresponding fundamental basis is composed of two cycles of length 2 and two cycles of length 3 leading to a fundamental cycle basis of total weight 10. However, a minimum weight basis of total weight 9 is given by the three outer cycles of length 2 and the central triangle.

However, Horton [Hor87] proved that computing a minimum weight basis with $\mathbb{Z}/2\mathbb{Z}$ coefficients can be done in polynomial time. His algorithm is based on the greedy algorithm over combinatorial structures called *matroids*.

5.2.1 The Greedy Algorithm

As a vector space, $Z(G)$ inherits a **matroid** structure. A matroid is indeed an abstraction of a vector space that only retains linear dependencies. It is defined by a **ground set** S (intuitively the set of vectors) and a nonempty **family of independent sets** $\mathscr{I} \subset 2^S$ that satisfies

- **the hereditary property:** $J \in \mathscr{I}$ and $I \subset J$ implies $I \in \mathscr{I}$, and
- **the exchange property:** $I, J \in \mathscr{I}$ and $|I| < |J|$ implies that $I \cup \{x\} \in \mathscr{I}$ for some $x \in J \setminus I$.

A **basis** is just a maximally independent set. By the exchange property, all the bases have the same cardinality. Matroid theory was introduced by Hassler Whitney (1935) and has many applications including combinatorial optimization, discrete geometry, etc. When the elements of the ground set are weighted, there is a famous *greedy algorithm* that determines a minimum weight basis. It works as follows: maintain an independent set starting from the empty set, and iteratively add an element x to the current set I if $I \cup \{x\}$ is independent and if x has minimum weight among such elements. The algorithm stops when no x can be found, *i.e.* when I is a basis. In practice, the elements of S are scanned in increasing order of weights, so that each time an x is found such that $I \cup \{x\}$ is independent it can be added to the current I. The whole set S is thus scanned only once during the algorithm.

Theorem 5.2.1. *The greedy algorithm returns a minimum weight basis.*

PROOF. Let $(x_1, x_2, \ldots, x_r)$ be the basis returned by the greedy algorithm, where x_i is the ith inserted element. By the choice of each element we have $|x_1| \leq |x_2| \leq \cdots \leq |x_r|$, where $|x|$ is the weight of x. Consider any other basis $(y_1, y_2, \ldots, y_r)$ indexed in non-decreasing order: $|y_1| \leq |y_2| \leq \cdots \leq |y_r|$. Suppose by way of contradiction that there is some index i such that $|y_i| < |x_i|$ and choose such i as small as possible. Then, by the exchange property we can find $y \in \{y_1, \ldots, y_i\}$ such that $\{x_1, \ldots, x_{i-1}, y\}$ is independent. Since $|y| \leq |y_i| < |x_i|$ this would contradict the choice of x_i. It follows that $|y_i| \geq |x_i|$ for all i, implying that $(x_1, x_2, \ldots, x_r)$ has minimum weight. □

Since the cycle space contains 2^r cycles, the greedy algorithm per se does not seem very efficient. In order to restrict the search of a new basis element at each step of the algorithm, Horton [Hor87] gave a characterization of the cycles that may belong to a minimum weight basis.

Lemma 5.2.2. *Suppose $b = c + d$ is a cycle of a basis B of $Z(G)$. Then either $B \setminus \{b\} \cup \{c\}$ or $B \setminus \{b\} \cup \{d\}$ is a basis.*

PROOF. If c and d were both in the linear span of $B \setminus \{b\}$, then so would b. □

Corollary 5.2.3. *Assuming positive weights, the cycles of a minimum weight basis are simple.*

PROOF. Suppose that b is a non-simple cycle of a minimum weight basis B. Then b can be written as the sum $b = c + d$ of two edge disjoint cycles. In particular, b is longer than c or d. By the preceding lemma, we can replace b by c or d in B to get a shorter basis, contradicting the minimality of B. □

Note: if some of the weights cancel, then basically the same proof shows the existence of a minimum weight basis with simple cycles only.

Lemma 5.2.4. *Let b be a cycle of a minimum weight basis. Let p and q be two edge disjoint paths such that $b = p \cdot q^{-1}$. Then p or q is a shortest path.*

PROOF. Let t be a shortest path from the common initial vertex of p and q to their common last vertex. With a little abuse of notation, we can write $b = p \cdot t^{-1} + t \cdot q^{-1}$. By Lemma 5.2.2, b must be no longer than $p \cdot t^{-1}$ or $t \cdot q^{-1}$, implying with Corollary 5.2.3 that either q or p is a shortest path. □

Corollary 5.2.5. *Let v be a vertex of a cycle b of a minimum weight basis. Then b decomposes into $p \cdot a \cdot q^{-1}$ where a is an arc and p, q are two shortest paths with v as initial vertex.*

PROOF. Consider the arc sequence $(a_1, a_2, \ldots, a_k)$ of b with v the origin vertex of a_1 and the target of a_k. Let i be the maximal index such that $(a_1, a_2, \ldots, a_i)$ is a shortest path. Then $b = (a_1, a_2, \ldots, a_i) \cdot a_{i+1} \cdot (a_{i+2}, \ldots, a_k)$ and the previous lemma implies that $(a_{i+2}, \ldots, a_k)$ is a (possibly empty) shortest path □

When there is a unique shortest path between every pair of vertices, this corollary allows us to reduce the number of scanned cycles at each addition step of the greedy algorithm to nm cycles, one for each (vertex, edge) pair. For the rest of this section, we assume uniqueness of shortest paths and discuss the general case in the next section. We denote by $\gamma_{v,e}$ the cycle obtained by connecting the endpoints of e with shortest paths to v. By the uniqueness of shortest paths, $\gamma_{v,e} = \gamma_{v,e}^T$ where T is the shortest path tree rooted at v.

Proposition 5.2.6. *Let $G = (V, E)$ be a connected graph with n vertices and m edges and let $r = 1 - n + m$ be the rank of its cycle space. A minimum weight basis of $Z(G)$ can be computed in $O(n^2 \log n + r^2 nm) = O(nm^3)$ time.*

PROOF. By Corollary 5.2.5, we can restrict the scan step of the greedy algorithm to the cycles $\gamma_{v,e}$ with $(v, e) \in V \times E$. For each vertex v, we compute a shortest path tree T in $O(n \log n + m)$ time using Dijkstra's algorithm. There are r nontrivial cycles of the form $\gamma_{v,e}^T$, each of size $O(n)$. Their computation and storage for all the vertices v thus requires $O\big(n(n \log n + m + rn)\big)$ time. They can be sorted according to their length in $O(rn \log(rn))$ time. In order to check if a cycle is independent of the current family of basis elements, we view a cycle as a vector in $(\mathbb{Z}/2\mathbb{Z})^E$. We use Gauss elimination to maintain the current family in row echelon form. This family has at most r vectors and testing a new vector against this family by Gauss elimination needs $O(rm)$ time. The cumulated time for testing independence is thus $O(r^2 nm)$. The whole greedy algorithm finally takes time

$$O\big(n(n \log n + m + rn) + rn \log(rn) + r^2 nm\big) = O(n^2 \log n + r^2 nm).$$

□

Note that the above scan can be further reduced by discarding the cycles $\gamma_{v,e}$ that are not simple. We can also decompose a cycle into a linear combination of a fixed fundamental basis associated to a tree. The decomposition of a cycle is just given by its trace over the chords of that tree. This allows to represent the current family of basis elements by a matrix of size $r \times r$ instead of $r \times m$. Further improvements were proposed [KMMP04, KMMP08, MM09], often based on randomization. In particular, the algorithm by Kavitha et al. [KMMP08] runs in $O(m^2 n + mn^2 \log n)$ time. Using integer coefficients rather than $\mathbb{Z}/2\mathbb{Z}$ gives a more general notion of cycle space. However, this space does not form a matroid in general and the greedy algorithm cannot be applied anymore. The status of the computation of a minimal weight cycle basis with integer coefficients is still unknown.

5.3 Uniqueness of Shortest Paths

The proof of Proposition 5.2.6 is based on the uniqueness of shortest paths. In fact, the proof can be adapted to show that the same algorithm works even if we do not assume that there is a unique shortest path between every pair of vertices. See [Hor87] or [Laz14, Lem. 1.6.7]. It may happen for other applications that we strongly need uniqueness to ensure correctness of the algorithms. We usually get the uniqueness by a *perturbation schema*, where the weight of each simple path is replaced by a slightly different one. Let P_{xy} be the set of simple paths with minimal unperturbed weight between vertices x and y. The perturbation should be such that P_{xy} contains a unique path of minimum perturbed weight. In other words, the aim is to get an order on each P_{xy} so that we can choose the smallest path as the unique shortest path. This can be achieved by adding an infinitesimal weight of the form (i) $\varepsilon^{c(e)}$ or (ii) $c(e)\varepsilon$ to every edge e, where $\varepsilon > 0$ is some arbitrarily small number and $c(e)$ is an appropriately chosen coefficient.

Using the exponential form (i) we can simply choose pairwise distinct edge coefficients, for example the edge indices, assuming that they are indexed from 1 to m. This way, distinct paths are perturbed by distinct polynomials in ε and get distinct weights for ε small enough. We can view the polynomials as bit vectors of length m where a 1 coordinate at index i indicates the presence of the monomial ε^i. The ordering in P_{xy} is simply the lexicographic ordering on the bit vectors. This perturbation schema would a priori require an extra $O(m)$ time for comparing path lengths. Cabello et al. [CCE13, Sec. 6.2] propose to reduce the comparison time to $O(\log m)$ using some sophisticated data structure. However, their algorithm assumes that two paths need to be compared only when they intersect along a common prefix.

We can avoid this restriction using the linear form (ii), that is when the weight of an edge e is perturbed by $c(e)\varepsilon$. The perturbation of a path is now ε times the sum of its edge coefficients. Choosing the edge coefficients such that there is a unique minimum weight sum in each P_{xy} is more tricky than for the form (i). Cabello et al. [CCE13, Sec. 6.1] propose the following random perturbation schema based on the Isolating Lemma of Mulmuley et al. [MVV87, Lem. 1].

Lemma 5.3.1 (Isolating –)**.** *Let $\mathscr{I}$ be an arbitrary family of subsets of $\{1,\dots,m\}$. For a vector $c=(c_1,c_2,\dots,c_m)$ of m integers and for $I\in\mathscr{I}$, we put $c(I)=\sum_{i\in I}c_i$. Choosing c uniformly at random in $\{1,\dots,M\}^m$, the probability that $c(I)$ is minimized by a unique $I\in\mathscr{I}$ is at least $1-m/M$.*

PROOF. We suppose that $\mathscr{I}$ contains at least two subsets, since otherwise the lemma is trivial. For $i\in\{1,\dots,m\}$ we set

$$\mathscr{I}_i^+=\{I\in\mathscr{I}\mid i\in I\}\quad\text{and}\quad\mathscr{I}_i^-=\{I\in\mathscr{I}\mid i\notin I\}$$

Suppose that none of $\mathscr{I}_i^+$ and $\mathscr{I}_i^-$ is empty. Note that the quantities $\min_{I\in\mathscr{I}_i^+}c(I)-c_i$ and $\min_{I\in\mathscr{I}_i^-}c(I)$ do not depend on c_i. Fixing all the coefficients c_j, for $j\neq i$, the constant $\min_{I\in\mathscr{I}_i^-}c(I)-(\min_{I\in\mathscr{I}_i^+}c(I)-c_i)$ equals c_i with probability at most $1/M$. It follows that $\min_{I\in\mathscr{I}_i^+}c(I)=\min_{I\in\mathscr{I}_i^-}c(I)$ holds with unconditional probability at most $1/M$. Hence, with probability at least $1-m/M$, $\min_{I\in\mathscr{I}_i^+}c(I)\neq\min_{I\in\mathscr{I}_i^-}c(I)$ for all i such that $\mathscr{I}_i^+$ and $\mathscr{I}_i^-$ are both nonempty. Consider a vector c such that this occurs and let $I_0\in\mathscr{I}$ for which $c(I_0)$ is minimum. Then, any other $J\in\mathscr{I}$ must differ from I_0 by some index i. If $i\in I_0$ and $i\notin J$ then $c(I_0)=\min_{I\in\mathscr{I}}c(I)=\min_{I\in\mathscr{I}_i^+}c(I)$ while $c(J)\geq\min_{I\in\mathscr{I}_i^-}c(I)$. Since $\min_{I\in\mathscr{I}_i^+}c(I)\neq\min_{I\in\mathscr{I}_i^-}c(I)$ we deduce $c(J)>c(I_0)$. Likewise, we again obtain $c(J)>c(I_0)$ if $i\notin I_0$ and $i\in J$. □

Lemma 5.3.2. *Choose for each of the m edges of an edge weighted graph G an integral coefficient in $\{1,\dots,m^4\}$ uniformly and independently at random. Consider the linear perturbation schema (ii) as described above. With probability at least $1-\frac{1}{2m}$, there is a unique shortest path between any pair of vertices.*

PROOF. For each pair $\{x,y\}$ of vertices, Let $\mathscr{I}_{xy}$ be the family of subsets of edge indices corresponding to the paths in P_{xy}. Applying Lemma 5.3.1 to $\mathscr{I}_{xy}$, we deduce that with probability at least $1-1/m^3$ there is a unique shortest path between x and y for the perturbed weights. There are $n\leq m$ vertices in G (we may assume that G is not a tree). Hence, the $\binom{n}{2}$ pairs of vertices are each connected by a unique shortest path with probability at least $1-\binom{n}{2}/m^3\geq 1-\frac{1}{2m}$. □

We shall turn to the computation of minimum bases on surfaces. We first extend the notion of cycle space to surfaces.

5.4 First Homology Group of Surfaces

5.4.1 Back to Graphs

First recall that the cycle space $Z(G)$ of a graph $G=(V,E)$ is the space of its Eulerian subgraphs. One can define such subgraphs thanks to the **boundary operator**. This operator δ_1 sends any edge to the mod 2 sum of its endpoints. In particular, if e is a loop-edge, $\delta_1 e=0$. By linear extension, δ_1 defines a linear map from the vector space $(\mathbb{Z}/2\mathbb{Z})^E$ of formal mod 2 sum of edges to the space $(\mathbb{Z}/2\mathbb{Z})^V$ of mod 2 sum of

vertices. Viewing a subgraph as a mod 2 sum of its edges, it is easily seen that Eulerian subgraphs correspond to the mod 2 sum of edges with empty boundary. In other words,

$$Z(G) = \ker \delta_1.$$

We define the **mod 2 abelianization** of a group A as the quotient $A/S(A)$ by the subgroup $S(A)$ generated by its squares. Note that $S(A)$ is normal and contains the **derived subgroup** $[A,A]$ generated by the **commutators** $[a,b] = aba^{-1}b^{-1}$. Indeed, one check that

$$[a,b] = (aba^{-1})^2 a^2 (a^{-1}b^{-1})^2 \quad \text{and} \quad asa^{-1} = s[s^{-1},a]$$

Hence, if s is a product of squares, so is any conjugate asa^{-1}. We can now relate the cycle space with the fundamental group of a graph thanks to the following mod 2 version of the Hurewicz theorem.

Proposition 5.4.1. *For any vertex v of a connected graph G the cycle space $Z(G)$ is isomorphic to the mod 2 abelianization of $\pi_1(G,v)$.*

PROOF. Denote by $\mathscr{L}$ the set of loops of G with basepoint v. Consider the map $\varphi : \mathscr{L} \to Z(G)$ defined by $(a_1, a_2, \dots, a_k) \mapsto \sum_{i=1}^{k} a_i$, where the coefficient in the sum are taken modulo 2. Adding or removing a spur in a loop does not change its image by φ. The map φ thus quotients to a morphism $\overline{\varphi} : \pi_1(G,v) \to Z(G)$. Let T be a spanning tree of G and let $C = E(G) \setminus E(T)$ be its set of chords. We know that $Z(G)$ is generated by the cycles $\{\gamma_e^T\}_{e\in C}$. Since $\varphi(\gamma_{v,e}^T) = \gamma_e^T$, the map $\overline{\varphi}$ is onto. Let $\gamma = \gamma_{v,e_1}^T \cdot \gamma_{v,e_2}^T \cdots \gamma_{v,e_k}^T$ be a representative of some element of $\pi_1(G,v)$ written over the basis $\{\gamma_{v,e}^T\}_{e\in C}$. Then $\varphi(\gamma) = \sum_{e\in C} n_e \gamma_e^T$ where n_e is the cumulated exponent of $\gamma_{v,e}^T$ in γ. Hence, the homotopy class of γ belongs to $\ker\overline{\varphi}$ if and only if all the n_e cancel. This is exactly saying that γ belongs to the subgroup $S(\pi_1(G,v))$ of $\pi_1(G,v)$. We thus have

$$Z(G) \simeq \pi_1(G,v)/\ker\overline{\varphi} = \pi_1(G,v)/S(\pi_1(G,v))$$

□

Exercise 5.4.2. Given a product w in the generators $\{x_1, x_2, \dots, x_r\}$ (and their inverses) of a group Γ, show that $w = x_1^{n_1} \cdot x_2^{n_r} \cdots x_1^{n_r} \cdot p$ where each n_i is the cumulated exponent of x_i in w and p is a product of commutators. (It might be useful to notice the relation $ba = ab[b^{-1}, a^{-1}]$.) Deduce that $S(\Gamma)$ is equal to the set of products whose cumulated exponents are all even.

5.4.2 Homology of Surfaces

The graph G of a combinatorial surface M has its own cycle space $Z(G)$. However, a topological surface may have distinct cellularly embedded graphs with non-isomorphic cycle spaces. In order to get a topologically invariant notion of cycle space, we further quotient $Z(G)$ by identifying cycles that bound together a subset of faces of M. More formally, let $C_2(M) := (\mathbb{Z}/2\mathbb{Z})^F$ be the vector space of subsets of the set F of faces of M. The elements of $C_2(M)$ are called **2-chains**. The boundary $\partial_2 f$ of a face $f \in F$ is the mod 2 sum of the edges of its facial walk. It is clearly a cycle of $Z(G)$, meaning that

$\partial_1\partial_2 f = 0$. This boundary ∂_2 extends linearly to a boundary operator $\delta_2 : C_2(M) \to Z(G)$. Two cycles $c, d \in Z(G)$ are said **homologous**, which we write $[c] = [d]$, if their mod 2 sum is the boundary of some 2-chain $\sigma \in C_2(M)$: $c - d = \partial_2\sigma$. We can now define the **first homology group** of M as the space of homology classes:

$$H_1(M) := \ker \delta_1 / \operatorname{Im} \delta_2.$$

The fact that this homology group is indeed a topological invariant is an immediate consequence of the invariance of the fundamental group and of the following mod 2 version of the Hurewicz theorem for surfaces.

Proposition 5.4.3. *For any vertex v of a connected map M the first homology group $H_1(M)$ is isomorphic to the mod 2 abelianization of $\pi_1(M, v)$.*

PROOF. Let G be the graph of M. As in the proof of Proposition 5.4.1, denote by $\mathscr{L}$ the set of loops of G with basepoint v and by $\varphi : \mathscr{L} \to Z(G)$ the mapping defined by $\varphi(a_1, a_2, \ldots, a_k) = \sum_{i=1}^{k} a_i$. The composition $[\varphi] : \mathscr{L} \to Z(G) \to H_1(M)$ is compatible with elementary homotopies in M. This is obvious for the addition or removal of a spur. If $\lambda \cdot p \cdot \mu \mapsto \lambda \cdot q \cdot \mu$ is an elementary homotopy with $p \cdot q^{-1}$ the facial walk of a face f, then $\varphi(\lambda \cdot p \cdot \mu) - \varphi(\lambda \cdot q \cdot \mu) = \partial_2 f \in \operatorname{Im} \partial_2$. Whence $[\varphi(\lambda \cdot p \cdot \mu)] = [\varphi(\lambda \cdot q \cdot \mu)]$ in $H_1(M)$. It follows that $[\varphi]$ descends to the quotient $\overline{\varphi} : \pi_1(M, v) \to H_1(M)$. On the other hand, homotopic loops in G are homotopic in M so that we have an onto morphism $\pi_1(G, v) \twoheadrightarrow \pi_1(M, v)$. We also know from the proof of Proposition 5.4.1 that the morphism $\pi_1(G, v) \twoheadrightarrow Z(G)$ is onto. We thus have two equal compositions $\pi_1(G, v) \twoheadrightarrow Z(G) \twoheadrightarrow H_1(M)$ and $\pi_1(G, v) \twoheadrightarrow \pi_1(M, v) \xrightarrow{\overline{\varphi}} H_1(M)$ implying that $\overline{\varphi}$ is onto.

It remains to prove that $\ker \overline{\varphi}$ is the subgroup $S(\pi_1(M, v))$ generated by the squares of $\pi_1(M, v)$ to conclude that $H_1(M) \simeq \pi_1(M, v)/\ker \overline{\varphi}$ is the mod 2 abelianization of $\pi_1(M, v)$. Since multiplication by 2 gives zero in $H_1(M)$, we have $S(\pi_1(M, v)) \subset \ker \overline{\varphi}$. For the reverse inclusion we consider a loop γ whose homotopy class is in $\ker \overline{\varphi}$, *i.e.* such that $[\varphi(\gamma)] = 0$. Hence, there must be a 2-chain $\sum_j f_j$ such that $\varphi(\gamma) = \sum_j \partial_2 f_j$. For each j, we choose a vertex v_j incident to f_j and we let p_j be the facial walk of f_j starting at v_j. Using the path γ^T_{v,v_j} from v to v_j in T we form the loop $\gamma_j := \gamma^T_{v,v_j} \cdot p_j \cdot (\gamma^T_{v,v_j})^{-1}$ with basepoint v. On the one hand, since $\varphi(\gamma_j) = \partial_2 f_j$, we have $\varphi(\gamma) = \varphi(\prod_j \gamma_j)$ in $Z(G)$. Equivalently, $\varphi(\gamma) + \varphi(\prod_j \gamma_j) = 0$. By Proposition 5.4.1, the homotopy class of $\gamma \cdot \prod_j \gamma_j$ is in $S(\pi_1(G, v))$. It is thus in $S(\pi_1(M, v))$ viewed as a loop in M. On the other hand, since each γ_j is contractible in M, the loops $\gamma \cdot \prod_j \gamma_j$ and γ are homotopic in M. It follows that the homotopy class of γ is in $S(\pi_1(M, v))$. □

Corollary 5.4.4. *Let M be a combinatorial surface of genus g without boundary. We have*

$$H_1(M) \simeq \begin{cases} (\mathbb{Z}/2\mathbb{Z})^{2g} & \text{if } M \text{ is orientable, and} \\ (\mathbb{Z}/2\mathbb{Z})^{g} & \text{otherwise.} \end{cases}$$

PROOF. If M is orientable, we know that its fundamental group as combinatorial presentation $\pi_1 \simeq < a_1, b_1, \ldots, a_g, b_g \mid [a_1, b_1] \cdots [a_g, b_g] >$. By Proposition 5.4.3, we have

$H_1(M) \simeq \pi_1/S(\pi_1)$. Since $[a_1,b_1]\cdots[a_g,b_g] \in S(\pi_1)$, we also have $\pi_1/S(\pi_1) \simeq F_{2g}/S(F_{2g})$ where $F_{2g} :=< a_1,b_1,\ldots,a_g,b_g \mid -> $ is the free group over $\{a_1,b_1,\ldots,a_g,b_g\}$. Now, it is easily seen that the mod 2 abelianization of a free group of rank r is the $\mathbb{Z}/2\mathbb{Z}$-vector space of dimension r, whence $H_1(M) \simeq F_{2g}/S(F_{2g}) \simeq (\mathbb{Z}/2\mathbb{Z})^{2g}$. A similar proof holds when M is non-orientable. □

5.5 Minimum Basis of the Fundamental Group of a Surface

Let M be a combinatorial surface with graph G. As in Section 5.1, we assume that the edges of G are positively weighted. Given a vertex v of M, a *minimum weight basis* of $\pi_1(M,v)$ is a set of loops with basepoint v whose homotopy classes form a basis of $\pi_1(M,v)$ and whose total weight is minimum. Erickson and Whittlesey [EW05] have proposed a simple algorithm to compute a minimum weight basis. We first describe how to formally cut M along a subgraph of G.

5.5.1 Dual Maps and Cutting

The **dual map** M^* of M is obtained by inverting the roles of vertices and edges in M. Its graph G^* is the dual graph of G. If H is a subgraph of G, we denote by H^* the subgraph of G^* induced by the edges dual to the edges of H. We also denote by $M\backslash\backslash H$ the map with boundary obtained by cut opening M along H. It boils down to double the edges of H, updating the rotation system of M to include these new edges. Equivalently, if one views M as a polygonal schema, *i.e.* as a gluing of polygons by pairwise identifications of their sides, cutting along H amounts to forbid the identification between the sides that correspond to edges in H. In the dual map, the effect is to delete the corresponding dual edges. Hence,

Lemma 5.5.1. *The adjacency graph of the faces of $M\backslash\backslash H$ is $G^* - E(H^*)$. In particular, the connected components of $M\backslash\backslash H$ and of $G^* - E(H^*)$ are in 1-1 correspondence.*

As usual, $E(H^*)$ designates the set of edges of H^*.

5.5.2 Homotopy Basis Associated with a Tree-Cotree Decomposition

Recall that a tree-cotree decomposition (T,D^*,C) of M is given by a spanning tree T of G, a spanning tree D^* of $G^* - E(T^*)$, and the complementary set of edges $C = E(G)\setminus(E(T)\cup E(D))$.

Lemma 5.5.2. *If (T,D^*,C) is a tree-cotree decomposition of M, then C contains $2-\chi(M)$ edges. In particular the cycle spaces of the graphs $T\cup C$ and $D^*\cup C^*$ have dimension $2-\chi(M)$*

PROOF. The trees T and D^* being spanning we have $|E(T)| = |V(M)|-1$ and $|E(D^*)|=|F(M)|-1$. Thanks to Euler formula, we can write

$$|V(M)|+|F(M)|-\chi(M)=|E(M)|=|E(T)|+|E(D^*)|+|C|=|V(M)|+|F(M)|-2+|C|,$$

whence $|C|=2-\chi(M)$. □

In analogy with the basis of the fundamental group of a graph associated with a spanning tree, we can associate a basis of the fundamental group of M with a tree-cotree decomposition.

Lemma 5.5.3. *Let v be a vertex of M, and let (T,D^*,C) be a tree-cotree decomposition of M. The set of loops $\{\gamma^T_{v,c}\}_{c\in C}$ is a basis of $\pi_1(M,v)$.*

PROOF. Since D^* is a tree, the gluing of faces of M along the edges of D is a disk. In other words, $M\setminus\setminus(T\cup C)$ is a disk. Hence, every edge $d\in E(D)$ cuts this disk into two disks. Choose one of those disks. Its boundary writes $(d,e_1,\dots,e_k)$ where each e_i is an edge of $T\cup C$. This boundary is obviously contractible. By inserting a round-trip to v in T at each vertex along this boundary, we see that $\gamma^T_{v,d}\cdot\gamma^T_{v,e_1}\cdots\gamma^T_{v,e_k}$ is contractible. This shows that $\gamma^T_{v,d}$ is in the span of $\{\gamma^T_{v,c}\}_{c\in C}$ since γ^T_{v,e_i} is contractible for every e_i in T. Now, since $\{\gamma^T_{v,e}\}_{e\in E(D)\cup C}$ is a (fundamental) basis of $\pi_1(G,v)$, it is also a generating set for $\pi_1(M,v)$. In turn this generating set is generated by $\{\gamma^T_{v,c}\}_{c\in C}$. Finally, by Lemma 5.5.2 we note that C contains the minimum number of elements required for a basis of $\pi_1(M,v)$. □

5.5.3 The Greedy Homotopy Basis

For each chord e of T, the loop $\gamma^T_{v,e}$ is a shortest loop through e with basepoint v and we define the weight of the edge e^* dual to e as

$$w(e^*)=|\gamma^T_{v,e}|,$$

where $|.|$ denotes the given weight function in G. We consider a maximum weight spanning tree K^* of $G^*-E(T^*)$ with respect to the weight function w, and we let C be the set of edges primal to the chords of K^* in $G^*-E(T^*)$. We thus have a tree-cotree decomposition (T,K^*,C) and the set of loops

$$\Gamma:=\{\gamma^T_{v,e}\}_{e\in C}$$

is the associated basis of $\pi_1(M,v)$. Following [EW05], we call Γ a **greedy homotopy basis**. The name comes from a greedy computation of the maximum spanning tree K^* which makes the loops in Γ appear in a greedy fashion. It results from Proposition 5.4.3 that the set of homology classes of the loops in Γ is a basis of $H_1(M)$. A **greedy factor** of a loop ℓ with basepoint v is any loop in Γ which appears with a non-zero coefficient in the decomposition of ℓ in this homology basis.

Lemma 5.5.4. *The weight $w(e^*)$ of any chord e of T in G is larger or equal to the weights (with respect to $|.|$) of the greedy factors of $\gamma^T_{v,e}$.*

PROOF. The set of chords of T is the disjoint union $E(K)\cup C$. If $e \in C$, then $\gamma^T_{v,e}$ is its own and unique greedy factor and the result is trivial. We now assume that $e \in E(K)$. We put $C_1 := \{c \in C \mid w(c^*) \leq w(e^*)\}$ and $C_2 := \{c \in C \mid w(c^*) > w(e^*)\}$. We consider the connected graph $K^*_e := G^* - (E(T^*) \cup C^*_1) = K^* + C^*_2$. We claim that $K^*_e - e^*$ is not connected. Otherwise, e^* would belong to a cycle of K^*_e. This cycle would contain an edge c^* in C^*_2 and exchanging e^* with c^* in K^* would produce a spanning tree with strictly larger weight, contradicting the maximality of K^*. It ensues from Lemma 5.5.1 that $M\setminus\setminus(T \cup C_1)$ is connected while $M\setminus\setminus(T \cup C_1 \cup \{e\})$ is not. Hence, e appears exactly once in the boundary of each component of $M\setminus\setminus(T \cup C_1 \cup \{e\})$. Considering the formal sum of the faces of one component and its image by the boundary operator, we obtain that $e + \kappa$ is 0-homologous for some chain κ with support in $T \cup C_1$. We conclude that the greedy factors of $\gamma^T_{v,e}$ are contained in $\{\gamma^T_{v,c}\}_{c\in C_1}$, as desired. □

Lemma 5.5.5. *Let ℓ be a loop with basepoint v in G. Any greedy factor of ℓ has weight at most $|\ell|$.*

PROOF. We consider ℓ as a loop of G and express its homotopy class in the free basis of $\pi_1(G, v)$ associated with the chords of T in G: $\ell \sim \gamma^T_{v,e_1} \cdot \gamma^T_{v,e_2} \cdots \gamma^T_{v,e_k}$. We assume this expression reduced, so that each e_i, $1 \leq i \leq k$, occurs at least once in ℓ. In particular, $|\ell| \geq w(e^*_i)$. Since any greedy factor of ℓ must occur as a greedy factor of some γ^T_{v,e_i}, we can apply Lemma 5.5.4 to e_i and conclude. □

We denote by $\gamma_1, \ldots \gamma_{|C|}$ the loops in the greedy homology basis Γ. Similarly to Lemma 5.1.2, we can easily show that

Lemma 5.5.6. *For any basis $\{\ell_i\}_{1\leq i\leq |C|}$ of $\pi_1(M, v)$, there exists a permutation τ of $\{1 \ldots |C|\}$ such that for each $i \in \{1 \ldots |C|\}$, the loop γ_i is a greedy factor of $\ell_{\tau(i)}$.*

It directly follows from the two preceding lemmas that

Proposition 5.5.7. *Any greedy homotopy basis is a minimum weight basis.*

In order to compute a greedy homotopy basis one needs to compute a shortest path tree and a maximum weight spanning tree. A shortest path tree of a graph with n vertices and m edges can be computed in $O(n \log n + m)$ time using Dijkstra's algorithm. Classic maximum (or minimum) weight spanning tree algorithms run[1] in $O(n \log n + m)$ time [Tar83]. Since a homotopy basis of a map of genus g has $O(g)$ loops, and since each loop of a greedy basis may have size $O(n)$ we obtain

[1] A faster $O(m)$ algorithm exists for embedded graphs. See for instance Sec. 3.1. of Éric Colin de Verdière course notes.

Theorem 5.5.8 ([EW05]). *Let M be a finite connected map of genus g without boundary with n vertices and m non-negatively weighted edges. Given a vertex v of M, a minimum weight basis of $\pi_1(M,v)$ can be computed in $O(n\log n + gn + m)$ time.*

5.6 Minimum Basis of the First Homology Group of a Surface

5.6.1 Homology Basis Associated with a Tree-Cotree Decomposition

In analogy with the fundamental cycle basis of a graph associated with a spanning tree, we can associate a basis of $H_1(M)$ with a tree-cotree decomposition.

Lemma 5.6.1. *Let (T, D^*, C) be a tree-cotree decomposition of M. The set of cycles $\{\gamma_c^T\}_{c\in C}$ is a basis of $H_1(M)$.*

PROOF. We can either reproduce the proof of Lemma 5.5.3, replacing contractible by 0-homologous, or directly apply Proposition 5.4.3. □

5.6.2 The Greedy Homology Basis

We again assume that the edges of the graph G of M are positively weighted. We also assume uniqueness of shortest path between each pair of vertices in G, see Section 5.3. Analogously to Section 5.2, we look for a basis of $H_1(M)$ such that the sum of the weights of the cycles in the basis is minimal. Since $H_1(M)$ is a vector space, the greedy matroidal algorithm of Section 5.2.1 remains valid as well as the characterization in Corollary 5.2.3 and 5.2.5 of the cycles in a minimum basis. For each vertex v of M we let T_v be a shortest path tree rooted at v. By Corollary 5.2.5, we can restrict the cycle scan in the greedy algorithm to simple cycles of the form $\gamma_{v,e} := \gamma_{v,e}^{T_v}$, one for each chord e of T_v. In fact, we can further restrict the scan to a subset of $O(g)$ candidate cycles per vertex.

Lemma 5.6.2. *The set of loops $\mathscr{L}_v = \{\gamma_{v,e} \mid e \in E(G)\setminus E(T_v)\}$ contains at most $3(1-\chi(M)) = O(g)$ distinct homology classes. Furthermore, we can select in $O(n\log n + m)$ time a subset $\mathscr{S}_v \subset \mathscr{L}_v$ of at most $3(1-\chi(M))$ loops that contains a homologous loop of minimal weight for each homology class in $\mathscr{L}_v$.*

PROOF. Following Section 5.5.1 we use a * superscript to denote duality. Put $K^* := G^* - E(T_v^*)$. Since T_v is a tree, it can be completed to a tree-cotree decomposition of M and it results from Lemma 5.5.2 that the cycle space $Z(K^*)$ has dimension $2-\chi(M)$. If $e_1^*,\dots,e_k^*$ are the edges incident to a vertex dual to a face f of M, then $\partial_2 f = \sum_i e_i = \sum_i \gamma_{v,e_i}$, so that $\sum_i \gamma_{v,e_i}$ is null-homologous. This sum can be restricted to $e_i \in E(K)$ because γ_{v,e_i} is null-homologous otherwise. It follows that $\gamma_{v,e}$ is also null-homologous whenever e^* is a pendant edge in K^*. We can delete recursively all the pendant edges in K^* since their corresponding cycle is null-homologous. We are left with a subgraph

K_1^* without degree one vertex and with the same cycle space as K^*. If two edges e^* and e'^* share a degree two vertex in K_1^* we also have that $\gamma_{v,e}$ and $\gamma_{v,e'}$ are homologous. It follows that the number of distinct homology classes is at most the number of maximal chains, *i.e.* of maximal paths with degree two internal vertices in K_1^*. This is also the number of edges of the graphs K_2^* obtained by contracting each such maximal chain to a single edge. Because each vertex of K_2^* has degree three or more, we have $2|E(K_2^*)| \geq 3|V(K_2^*)|$ by double counting of the vertex-edge incidences. On the other hand,

$$2-\chi(M) = \dim Z(K^*) = \dim Z(K_1^*) = \dim Z(K_2^*) = 1-|V(K_2^*)|+|E(K_2^*)|$$

It ensues that $|E(K_2^*)| \leq 3(|E(K_2^*)|-|V(K_2^*)|) = 3(1-\chi(M))$ as desired. In practice, we first compute T_v and the distance of each vertex to the root v in $O(n\log n+m)$ time using Dijkstra's algorithm. For any edge e of M, the length of $\gamma_{v,e}$ can then be computed in constant time. We recursively remove the pendant edges of K^* and traverse each maximal chain of the resulting graph K_1^* in linear time, only keeping in $\mathscr{S}_v$ the loop $\gamma_{v,e}$ corresponding to the traversed edge e^* if the loop has minimum weight in the maximal chain. □

The greedy matroidal algorithm requires to test if a loop is homologically independent of the already selected loops. To this end we consider a fixed homology basis $\mathscr{B} := \{\gamma_c^T\}_{c\in C}$ associated with some tree-cotree decomposition (T, D^*, C).

Lemma 5.6.3. *We can compute the homology coordinates with respect to $\mathscr{B}$ of each of the loops in $\mathscr{S}_v$ in $O(gm)$ total time.*

PROOF. We first compute for each edge e of M, the coordinates of γ_e^T with respect to $\mathscr{B}$. This can be done in $O(gm)$ time for all the edges in D by a simple traversal of the dual tree D^*. We then traverse the shortest path tree T_v from its root v in order to compute for each vertex x the homology coordinates with respect to $\mathscr{B}$ of the loop $\gamma_v(x) := \gamma_{v,x}^{T_v} \cdot \gamma_{x,v}^T$ composed of the two $(x-v)$-paths in T_v and T respectively. The traversal needs $O(gn)$ time, spending $O(g)$ time per vertex to compute the coordinates of $[\gamma_v(x)] = [\gamma_v(y)]+[\gamma_{yx}^T]$ using the predecessor y of x in T_v. The coordinates of any $\gamma_{v,e}$ in $\mathscr{S}_v$ can now be decomposed into the sum of the coordinates of $\gamma_v(x)$, γ_e^T and $\gamma_v(y)$ where x, y are the endpoints of e. It thus takes $O(g)$ time to compute the coordinates of any loop in $\mathscr{S}_v$ and the whole computation needs $O(gm)$ time. □

Theorem 5.6.4 ([EW05])**.** *Let M be a finite connected map of genus g with n vertices and m weighted edges. A minimum weight basis of $H_1(M)$ can be computed in $O(n^2\log n+gnm+g^3n)$ time.*

PROOF. We can select $O(gn)$ loop candidates for the minimal weight basis and compute their weights in $O(n^2\log n+nm)$ time according to Lemma 5.6.2. Their homology coordinates with respect to $\mathscr{B}$ is computed in $O(gnm)$ time following Lemma 5.6.3. After sorting the $O(gn)$ candidate loops according to their weight in $O(gn\log n)$ time, the greedy algorithm consists in scanning the candidate loops in

increasing order, keeping the scanned loop in the minimal basis if it is homologically independent of the previously selected loops. This last test can be answered in $O(g^2)$ time using Gauss elimination to maintain the $O(g)$ selected loops in row echelon form. The whole scan thus takes $O(g^3 n)$ time. Summing up all the steps we may conclude the theorem. □

When $g = o(n^{1/3})$, a faster $O(g^3 n \log n)$ algorithm was obtained by Borradaile et al. [BCFN16]. It combines the approach of Kavitha et al. [KMMP08] for the minimum cycle basis with the use of a certain cyclic covering to compute each cycle of the minimum weight basis.

6

Homology

Contents

As we have seen in the previous lecture, as the dimension of the space under study increases, most topological problems become very quickly undecidable. In this lecture, we investigate one of the only topological notions that remains computable no matter the dimension: homology.

6.1 Complexes

In order to talk about algorithms, we first explain how to describe and manipulate topological spaces of arbitrary dimension. Following the path that we used for surfaces, we will build complicated spaces by gluing together fundamental blocks, called simplices. The resulting object is generally called a complex. However, there are many different ways in which this can be done, each of which having advantages and disadvantages.

We will focus on two closely related concepts: we first introduce simplicial complexes, which are very simple to define but sometimes cumbersome to use, which is why we generalize them slightly to Δ-complexes.

An **affine simplex** in dimension n is the convex hull of $n+1$ affinely independent points in $\mathbb{R}^p$ for some big enough p and a **simplex** is the topological space defined by this affine simplex. The points are called the **vertices** of the simplex and a **face** of a simplex σ is a simplex defined by a subset of the vertices of σ. A **simplicial complex** is a collection X of simplices such that every face of a simplex in X is also in X, and any two simplices of X intersect in a common face, possibly empty. The **dimension** of a simplicial complex is the maximal dimension of its simplices.

For technical reasons, we will require *orientations* on simplices. For a simplex σ with vertices $s_0, \dots, s_n$, we consider two permutations on the vertices s_i to be equivalent if they have the same parity, and an orientation is such an equivalence class. We will use the notation $[s_0, \dots, s_n]$ to denote a simplex endowed with the orientation induced by the permutation $(s_0, \dots, s_n)$. For an oriented simplex σ, we denote by $-\sigma$ the same simplex with the opposite orientation. In the rest of these notes, we will always consider that the simplices in a simplicial complex are oriented in an arbitrary fashion.

The notion of simplicial complex is a bit awkward, because it requires some ambient space, and more restrictive than the notions of triangulations that we saw for surfaces and 3-manifolds: it allows gluings between different simplices, but not identifications within a single simplex: for example the weird triangulation of Exercise 3.6 in the lecture on knots and 3-manifolds is not a simplicial complex. The notion of **Δ-complex** is a generalization of simplicial complexes that allows us to identify the faces of a collection of simplices pretty much as we want. An n-simplex is the image of an affine simplex by a homeomorphism. A Δ-complex is the last space in an inductively defined sequence of topological spaces $X^{(0)} \subseteq \dots \subseteq X^{(n)} = X$, where each space $X^{(k)}$ is called the k-skeleton of X. For each integer $k > 0$, we inductively construct the k-skeleton $X^{(k)}$ by attaching a set of k-simplices to the $(k-1)$-skeleton $X^{(k-1)}$. Each k-simplex Δ_k is attached by a **gluing map** $\sigma : \partial\Delta_k \to X^{(k-1)}$ that maps the interior of each face of Δ_k homeomorphically to the interior of a simplex in $X^{(k-1)}$ of the same dimension. Such maps are called **cellular**.

For example, a triangulated map is a 2-dimensional Δ-complex. If the underlying graph is a simple graph this can even be realized as a simplicial complex. Note however that some maps that we used for surfaces (for example the polygonal schemes) are complexes which are still not Δ-complexes. They fit within a further generalization which is the notion of **polyhedral complexes**, which allows the building blocks to be arbitrary polyhedra and not just simplices. We will not define it in this course, but everything homological works the same for them. An even further generalization leads to the notion of a **CW-complex**, where the gluing maps are not required to be cellular. We will also not define these, and just say that everything homological works almost the same, except that some care must be taken when handling attaching maps.

6.2 Homology

6.2.1 Chain complexes

The homology of a Δ-complex is obtained by first defining a *chain complex* out of a Δ-complex, and then taking the homology of this chain complex. This chain complex depends on the choice of a *coefficient ring* which we will denote by R in these notes. The correct level of generality to study homology will involves modules over this ring R, but for the reader not very acquainted with commutative algebra, only considering the cases where R is the set of integers $\mathbb{Z}$ (leading to finitely generated abelian groups) or the p-element field $\mathbb{Z}_p$ for p prime (leading to vector spaces over $\mathbb{Z}_p$) is enough for intuition and for most practical purposes.

The space of **k-chains** of a Δ-complex X is the set of formal linear combinations over R of its oriented simplices of dimension k, i.e., a k-chain is a function $\alpha : X_k \to R$ where X_k is the set of simplices of dimension k in X. A k-chain is generally written as a formal sum $\sum_i \alpha_i \Delta_{k,i}$ where $\Delta_{k,i}$ is the ith oriented k-simplex in X and $\alpha_i = \alpha(\Delta_{k,i})$. Morally, α describes how many times one picks each simplex in the complex X. We denote by $C_k(X)$ the space of k-chains of a complex X. It is isomorphic (as a module) to R^{n_k} where n_k denotes the number of k-simplices in X.

The **boundary** of an oriented simplex is defined by

$$\partial_k[s_0,\dots,s_k] = \sum_{i=0}^{n} (-1)^i [s_0,\dots,\hat{s_i},\dots,s_k]$$

where $\hat{s_i}$ denotes the omission of the vertex s_i. Thus the boundary of a k-chain is a $(k-1)$-chain, and by linear extension the boundary operator ∂_k can be defined on the set of k-chains $C_k(X)$ in the following way:

$$\begin{array}{rcl} \partial_k : C_k(X) & \to & C_{k-1}(X) \\ \sum_i \alpha_i \Delta_{k,i} & \mapsto & \sum_i \alpha_i \partial_k(\Delta_{k,i}). \end{array}$$

The key relation in homology is that the boundary of a boundary is empty:

Lemma 6.2.1. $\partial_{k-1} \circ \partial_k = 0$.

PROOF. By linearity, it is enough to prove that the boundary of the boundary of a k-simplex is the empty $(k-2)$-chain. This is a matter of computation:

$$\begin{array}{rl} \partial_{k-1}[s_0,\dots,\hat{s_i},\dots,s_k] & = \\ \sum_{j<i} (-1)^j [s_0,\dots,\hat{s_j},\dots,\hat{s_i},\dots s_k] - \sum_{j>i} (-1)^j [s_0,\dots,\hat{s_i},\dots,\hat{s_j},\dots s_k] & \end{array}$$

and thus

$$\partial_{k-1}\circ\partial_k[s_0,\dots,s_n] \quad =$$
$$\sum_{i=0}^{k}\sum_{j<i}(-1)^{i+j}[s_0,\dots,\hat{s_j},\dots,\hat{s_i},\dots s_k]-\sum_{i=0}^{k}\sum_{j>i}(-1)^{i+j}[s_0,\dots,\hat{s_i},\dots,\hat{s_j},\dots s_k] \quad =0$$

since both sums are equal. □

Thus, for an n-dimensional Δ-complex X, we have a sequence of boundary morphisms linking the chain groups:

$$0\to C_n(X)\xrightarrow{\partial_n} C_{n-1}(X)\xrightarrow{\partial_{n-1}}\dots\xrightarrow{\partial_2} C_1(X)\xrightarrow{\partial_1} C_0(X)\to 0,$$

where $\partial_{k-1}\circ\partial_k=0$ and 0 denotes the trivial group. Such a sequence is called a **chain complex**.

6.2.2 Simplicial homology

The space of **k-cycles**, denoted by $Z_k(X)$, is the space of simplices without boundary, i.e., the kernel of the morphism ∂_k. The space of **k-boundaries**, denoted by $B_k(X)$ is the image of ∂_{k+1}. Since $\partial_{k-1}\circ\partial_k=0$, we have $B_k(X)\subseteq Z_k(X)$, and this allows us to define the **kth homology group** $H_k(X)$ as the quotient

$$H_k(X)=Z_k(X)/B_k(X)=\ker\partial_k/\operatorname{Im}\partial_{k+1}.$$

The collection of all the homology groups $H_k(X)$ is usually denoted by $H_*(X)$.

Obviously, this definition depends heavily on the Δ-complex we consider, but it turns out that the homology groups are invariant under homeomorphism.

Theorem 6.2.2. *If X and Y are homeomorphic Δ-complexes, then $H_*(X)=H_*(Y)$.*

One naive way to establish it could be to follow the same line of thought as when we proved the invariance of many topological properties for surfaces:

PROOF. (False!) A Δ-complex X is a **refinement** of another Δ- complex Y if there is a homeomorphism from Y to X mapping any simplex of Y to a subcomplex of X. If X is a refinement of Y, one can easily prove that $H_*(X)=H_*(Y)$. Thus, if we can prove that any two homeomorphic Δ-complexes have a common refinement, this will prove the theorem. This is the (in)famous **Hauptvermutung** which we alluded to in the lecture notes for surfaces and 3-manifolds, which turns out to be *false* for dimensions four of higher. □

In order to circumvent the difficulty in this false proof, a more general formulation of homology, called **singular homology** has been introduced, which gives identical results for Δ-complexes, but can also be applied to spaces that have no triangulations. We will not delve into these technicalities and refer to Hatcher [Hat02] for the appropriate background on singular homology and the equivalence with the simplicial homology under study here.

6.2.3 Examples and the question of the coefficient ring

Morally, the homology groups count the number of holes in each dimension. But the situation is more subtle due to the torsion that may appear in the homology groups. We illustrate this on a few examples.

In order to compute the homology of a surface, the first step is to describe this surface as a Δ-complex. While, this can be definitely be done, this easily gets a bit unwieldy so we will cheat a bit, and work instead directly with polygonal schemes (i.e., with a polyhedral complex). The reader can verify that it gives the same result.

Let us start with the orientable surface S of genus g over the coefficient ring $\mathbb{Z}$, we take a system of loops made of $2g$ loops so that the resulting complex has 1 face, 1 vertex and $2g$ edges. Then the boundary of the vertex is trivial (as is always the case), but the boundary of the edges as well, since every edge is a loop. And in the boundary of the face, every edge appears once with each orientation, so they cancel out and the boundary is trivial as well. Thus the computation ends up being trivial and we have $H_0(S) = \mathbb{Z}$, $H_1(S) = \mathbb{Z}^{2g}$ and $H_2(S) = \mathbb{Z}$. We observe that the 1-dimensional homology is the *abelianization* of the fundamental group, which we already saw in the lecture notes on minimum weight bases (Proposition 4.2). It is a general fact, true in any dimension, known as the Hurewicz Theorem (see Hatcher [Hat02, Section 4.2]).

Now, for the non-orientable surface S of genus g, let us pick a system of loops corresponding to the polygonal scheme $a_1 \dots a_g \bar{a}_1 \dots \bar{a_{g-1}} a_g$, which (exercise) is the non-orientable surface of genus g. The boundary of the vertex and the edges is still trivial, but now the boundary of the face f is not, since a_g appears twice with the same orientation. Thus we have $\partial f = 2a_g$. Thus there are no non-trivial 2-cycles. The space of 1-dimensional cycles is generated by $a_1, \dots a_g$ and the space of 1-dimensional boundaries is generated by $2a_g$, thus $H_1(S)$ is isomorphic to $\mathbb{Z}^{g-1} \oplus \mathbb{Z}_2$. Thus there is a 1-dimensional "hole" that disappears when taken twice! Once again, one can verify that $H_1(S)$ is the abelianization of $\pi_1(S)$.

Exercise 6.2.3. Verify that the homology group is the same when computed with a canonical polygonal scheme $a_1 a_1 a_2 a_2 \dots a_g a_g$.

This is also a good illustration of the role of the coefficient ring. The reader can verify that for the orientable surface of genus g, taking the ring $\mathbb{Z}_2$ instead of $\mathbb{Z}$ yields $H_0(S) = \mathbb{Z}_2$, $H_1(S) = \mathbb{Z}_2^{2g}$ and $H_2(S) = \mathbb{Z}_2$, so it makes virtually no difference. But for the non-orientable surface, we obtain $\partial f = 2a_g = 0$ since $2 = 0$ in $\mathbb{Z}_2$. Thus there is a 2-dimensional cycle, and the homology groups are now $H_0(S) = \mathbb{Z}_2$, $H_1(S) = \mathbb{Z}_2^g$ and $H_2(S) = \mathbb{Z}_2$. In some sense, the coefficient ring $\mathbb{Z}_2$ prevents us from seeing the torsion that was detected with $\mathbb{Z}$.

Finally, let us try with the ring $\mathbb{Q}$. Now, $\partial f = 2a_g$ is non-trivial, but $\mathbb{Q}^g/2\mathbb{Q}$ is isomorphic to $\mathbb{Q}^{g-1}$, so we have yet another result: similarly the ring $\mathbb{Q}$ does not see the torsion. In some sense, the homology over $\mathbb{Z}$ is the one that contains the most information, this can be formalized in the *universal coefficient theorem* (see Hatcher [Hat02, Section 3.A]), which provides somewhat intricate algebraic constructions to deduce the homology groups over any ring from those over $\mathbb{Z}$.

6.2.4 Betti numbers and Euler-Poincaré formula

When the homology is taken with the coefficient ring $R = \mathbb{Z}$, the homology groups are finitely generated abelian groups, which can be decomposed as a product of cyclic groups:

$$H_k(X) = \mathbb{Z}^{\beta_k(X)} \times \prod_i (\mathbb{Z}/d_i\mathbb{Z})$$

for some integers β_k and $1 \leq d_1 \leq d_2 \leq \ldots \leq d_m$ where each integer d_i is a divisor of its successor d_{i+1}. Furthermore this decomposition is unique. The **rank** $\beta_k(X)$ of the free component of $H_k(X)$ is called the ***k*th Betti number** of X. The following formula is a wide-reaching generalization of the Euler formula we saw for planar and surface-embedded graphs.

Theorem 6.2.4 (Euler-Poincaré formula)**.** *Let X be a finite Δ-complex of dimension n, and let n_i be the number of simplices in dimension i, then*

$$\sum_{i=0}^{n} (-1)^i \beta_i(K) = \sum_{i=0}^{k} (-1)^i n_i.$$

PROOF. We have $\beta_i(X) = \text{rank}(H_i(X)) = \text{rank}(Z_i(X)) - \text{rank}(B_i(X))$, and by the rank formula, $\text{rank}(Z_i(X)) = n_i - \text{rank}(\text{Im}(\partial_i))$, thus

$$\begin{aligned} \sum_{i=0}^{k} (-1)^i \beta_i(K) &= \sum_{i=0}^{k} (-1)^i (n_i - \text{rank}\,\text{Im}(\partial_i) - \text{rank}\,\text{Im}(\partial_{i+1})) \\ &= \sum_{i=0}^{k} (-1)^i n_i - (-1)^k \text{rank}\,\text{Im}(\partial_{n+1}) - \text{rank}\,\text{Im}(\partial_0) \end{aligned}$$

which concludes the proof since the boundaries ∂_0 and ∂_{n+1} are empty. □

The quantity $\chi(X) = \sum_{i=0}^{n} (-1)^i \beta_i(K) = \sum_{i=0}^{k} (-1)^i n_i$ is called the **Euler characteristic** of X. Since the homology groups are a topological invariant, so is the Euler characteristic. The reader can cross-check with the examples in the previous subsection that for the case of graphs cellularly embedded on surfaces, we recover the Euler formula.

Remark: One could also define the Betti numbers as the alternate sum of the ranks of the homology groups obtained when taking a field of characteristic zero for the coefficient ring. By the aforementioned universal coefficient theorem, the two definitions coincide, but note that the hypothesis of characteristic zero is necessary here, as the computation of the $\mathbb{Z}_2$ homology groups of non-orientable surfaces illustrate.

6.2.5 Homology as a functor

We saw that the fundamental group of a surface not only associates a group to the surface, or more generally a topological space, but also associates a group morphism to every continuous maps: it is a *functor* from the category of topological spaces to the

category of groups. The same can be said for the homology, which maps the category of topological spaces to the one of modules, or vector spaces if the coefficient ring is a field. We will see how it works for the restricted case of Δ-complexes. A **simplicial map** $f : K \to L$ between two complexes K and L is a map that sends the vertices of K to the vertices of L and the simplex on the vertices $s_0, \dots, s_k$ to the simplex on the vertices $f(s_0, \dots, s_k)$. An example of a simplicial map is the inclusion map for a Δ-complex K included in another Δ-complex L.

One would like to extend by linearity a simplicial map into a map on the chains of the complex, but the orientations get in the way. Thus we define, for a simplicial map f, another map $f_\#$ that maps $[s_0, \dots, s_k]$ to $[f(s_0), \dots, f(s_k)]$ if the restriction of f to $\{s_0, \dots, s_k\}$ is injective, and 0 otherwise. Now this map can be extended by linearity to the chains of the complexes K and L, and it verifies $\partial_k \circ f_\# = f_\# \circ \partial_k$, thus it is a **morphism of chain complexes**. This property ensures that the maps $f_\#$ can be quotiented by the boundary groups: indeed, $f_\#(a + \partial\, b) = f_\#(a) + \partial(f_\#(b))$, so that the image of a and $a + \partial\, b$ is the same when quotiented by the boundary space. This allows to define a map $f_\# : H_*(K) \to H_*(L)$, which can also be denoted by $H_*(f)$.

Exercise 6.2.5. Check that $H_*(Id_K) = Id_{H_*(K)}$ and that if $f : K \to L$ and $g : L \to M$ are two simplicial maps, then $H_*(g) \circ H_*(f) = H_*(g \circ f)$. This property is called the **covariance** of the homology functor.

6.3 Homology computations

6.3.1 Over a field

One of the perks of homology is that it is a topological invariant living in the realm of modules, vector spaces or finitely generated abelian groups (depending on the choice of the coefficient rings). Unlike general groups where most problems are undecidable, these algebraic structures are very convenient to exploit in terms of computation. For instance, when the coefficient ring is a field like $\mathbb{Z}_2$, the spaces of chains, boundaries, cycles, and thus the homology groups are all vector spaces. In particular, computing the homology groups in this case amounts to computing kernels and images of explicit operators (the boundaries), and this boils down to linear algebra. For example, using Gaussian elimination, one can easily compute the homology groups in polynomial time in this case, and faster techniques for matrix multiplication allow us to do this even faster.

6.3.2 Computation of the Betti numbers: the Delfinado-Edelsbrunner algorithm

In order to compute the Betti numbers, there is a conceptually simple algorithm due to Delfinado and Edelsbrunner [DE95] which allows to bypass the use of linear algebra in low dimensions. The idea is to add the simplices of a Δ-complex one by one, i.e., to consider a Δ-complex K as a sequence of inclusions (technically, a **filtration**) $K_1 \subseteq K_2 \dots \subseteq K_m = K$, and to compute the Betti numbers incrementally.

Proposition 6.3.1. *Let K and K' be two Δ-complexes such that $K' = K \cup \sigma$ where σ is a k-simplex. If the boundary of σ in K' is a boundary in K, we have*

$$\beta_i(K') = \begin{cases} \beta_i(K)+1 \text{ if } i = k \\ \beta_i(K) \text{ otherwise.} \end{cases}$$

Otherwise,

$$\beta_i(K') = \begin{cases} \beta_i(K)-1 \text{ if } i = k-1 \\ \beta_i(K) \text{ otherwise.} \end{cases}$$

PROOF. Let us denote with a prime the objects related to K'. The chain complexes of K and K' are identical except for the part $C_k \to^{\partial_k} C_{k-1}$, thus $\beta'_i = \beta_i$ for $i \neq k, k-1$.

If $\partial'_k \sigma$ is a boundary in K, i.e., $\partial'_k \sigma \in \operatorname{Im} \partial_k$, then $\operatorname{Im} \partial'_k = \operatorname{Im} \partial_k$. Thus $\beta'_{k-1} = \beta_{k-1}$, and

$$\operatorname{rank} \ker \partial'_k = \operatorname{rank} C'_k - \operatorname{rank} \operatorname{Im} \partial'_k = \operatorname{rank} C_k + 1 - \operatorname{rank} \operatorname{Im} \partial_k = \operatorname{rank} \ker \partial_k + 1.$$

Thus, $\beta'_k = \beta_k + 1$.

In the other case, we have $\operatorname{rank} \operatorname{Im} \partial'_k = \operatorname{rank} \operatorname{Im} \partial_k + 1$ and $\ker \partial'_k = \ker \partial_k$, which gives similarly $\beta'_{k-1} = \beta_{k-1} - 1$ and $\beta'_k = \beta_k$. □

This proposition allows to compute the Betti numbers of a Δ-complex inductively, provided one can test whether the boundary of each newly added simplex was already a boundary. In low dimensions, this is easy: a 0-dimensional complex, i.e., a vertex has no boundary, and a 1-dimensional complex, i.e., an edge, has two vertices as its boundary. These were already a boundary if and only if they are in the same connected component of K, which can be tested easily (or not so easily but very efficiently using a Union-Find data structure). Some extensions to 2 and 3-dimensional complexes embedded in $\mathbb{S}^3$ are discussed in the article of Delfinado and Edelsbrunner [DE95], and for the general case this test can be done using the linear algebraic machinery alluded to above.

6.3.3 Over the integers: the Smith-Poincaré reduction algorithm

When the coefficient ring is not a field, one can still compute the homology groups, but this requires slightly more advanced techniques, which we now introduce in the paradigmatic case of the integers $\mathbb{Z}$. The **Smith-Poincaré reduction algorithm** is a variant of Gaussian elimination, tailored to deal with the integers instead of the reals.

The **Smith normal form** of an $r \times c$ integer matrix M is the description of M as a product $M = S\tilde{M}T$, where S is an invertible integral $r \times r$ matrix, T is an invertible integral $c \times c$ matrix and $\tilde{M}$ is an integral $r \times c$ matrix with only diagonal coefficients and each coefficient is a multiple of the previous one (some diagonal coefficients can be zero, but by this condition they have to be the last one). Throughout this paragraph, we will use without mention the well known connections between the multiplication by invertible matrices and the elementary operations on rows and columns. Note the similarity with Gaussian elimination, which consists of a similar factorization but without the constraint that the matrices are integral.

Let us assume that we put all the boundary operators ∂_k in Smith normal form, with diagonal elements $d_{k_1},\dots,d_{k_{m_k}}$, then the boundary, cycle, and homology groups are as follows: $Z_k=\mathbb{Z}^{n_k-m_k}$, $B_k=\mathbb{Z}^{m_{k+1}}$ and

$$H_k=\mathbb{Z}^{n_k-m_k-m_{k+1}}\oplus\bigoplus_{i=1}^{m_{k+1}}\mathbb{Z}_{d_{k_i}},$$

where it is understood that $\mathbb{Z}_1$ is trivial and is to omitted from this decomposition.

To prove the existence of a Smith normal form, we start with the following preparatory lemma:

Lemma 6.3.2. *There exist integral matrices S' and T' such that $M=S'\tilde{M}T'$ and in $\tilde{M}=(m'_{ij})$, all the m'_{ij} are multiple of m'_{11}.*

PROOF. We can assume that M is non-empty, otherwise the lemma is trivial. Let m_{ij} be the coefficient with the smallest absolute value. The proof is an induction on this absolute value.

If all the other coefficients are a multiple of m_{ij} (which includes the base case of the induction $m_{ij}=1$), we can put it in the top-left corner using permutations of the rows and the columns, which translate into permutation matrices for S and T.

Otherwise, there is some m_{kl} that is not a multiple of m_{ij}. If $k=i$ or $j=l$, doing the Euclidean division of $m_{kl}=\lambda m_{ij}+\alpha$, and subtracting λ times the ith/jth row/column to the kth/lth row/column, we have $\alpha<m_{ij}$ and we proceed by induction.

Finally, if all the coefficients on the ith row and the kth column are multiples of m_{ij}, but some m_{kl} persists in not being one, then we have $m_{il}=\lambda m_{ij}$ for some λ. Subtracting $\lambda-1$ times the jth column to the lth column, m_{il} gets transformed into α, and either $|m_{kl}|<\alpha$ and we proceed by induction, or we are in the previous case. □

We can now prove the theorem

Theorem 6.3.3. *Every integral matrix can be decomposed in a Smith normal form.*

PROOF. By the previous lemma, we can transform the top left element into one that divides all the other ones. Then, this element can be used to kill all the non-diagonal elements in the first row and column, and to obtain the matrix

$$\begin{pmatrix} m_{11} & 0 & \dots & 0 \\ 0 & & & \\ \dots & & M' & \\ 0 & & & \end{pmatrix}$$

where all the elements of M' are multiple of m_{11}. Then one can further reduce M' by induction. □

Exercise 6.3.4. Show that the Smith normal form of a matrix is unique.

The proof above yields an easily implementable algorithm to compute the Smith normal form, and thus the integral homology groups of a Δ-complex. However, the complexity of this algorithm is not so good: the size of the integers involved in the computations may easily explode – it is often hidden under the rug, but the same problem arises with the usual Gaussian elimination, which can be circumvented using the Bareiss algorithm (see for example the book of von zur Gathen and Gerhard [VZGG13]). More careful algorithms can be used to control the size of the integers and compute a Smith normal form in polynomial time, we refer to the survey of Dumas et al. [DHSW03].

7

Persistent Homology

Contents

The theory of persistent homology developed from 2000, motivated by practical problems related to approximation and reverse engineering [Rob99, ELZ00]. The main objective is to infer the topology of an object given by a finite cloud of points that approximates the object. In a typical application one is given a sampling P of the surface S of a manufactured object captured by some probing device. The question is to recover topological invariants (e.g., the number of connected components) of S with the sole knowledge of P. Although S and P may look very different, their ε-neighborhood have a similar topology for an appropriate range of ε. Recall that the ε-neighborhood of an object is the union of balls of radius ε centered at every point of the object. See Figure 7.1. This crucial observation is the basis of persistent homology. Since the correct range of ε is unknown and depends on the density of the sampling with respect to S, one is led to study the topology of the whole sequence of ε-neighborhoods of P for ε ranging from zero to infinity. See Figure 7.2. Note that $\varepsilon < \eta$ implies the inclusion of the ε-neighborhood in the η-neighborhood. Such a nested sequence of spaces is called a **filtration**. By applying the homology functor, each inclusion $X \subset Y$ in the filtration induces a linear map $H_*(X) \to H_*(Y)$. The idea

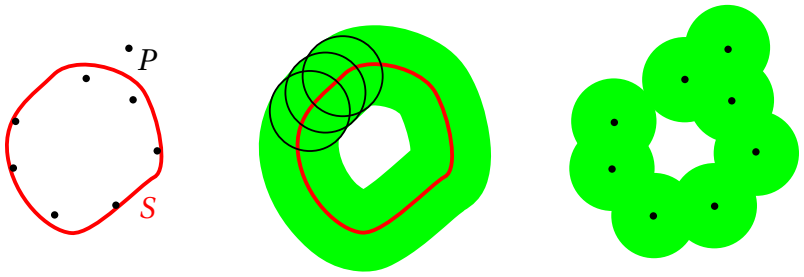


Figure 7.1: Left, an approximate sampling of a curve S. Middle, the ε-neighborhood of S. Right, the ε-neighborhood of P.

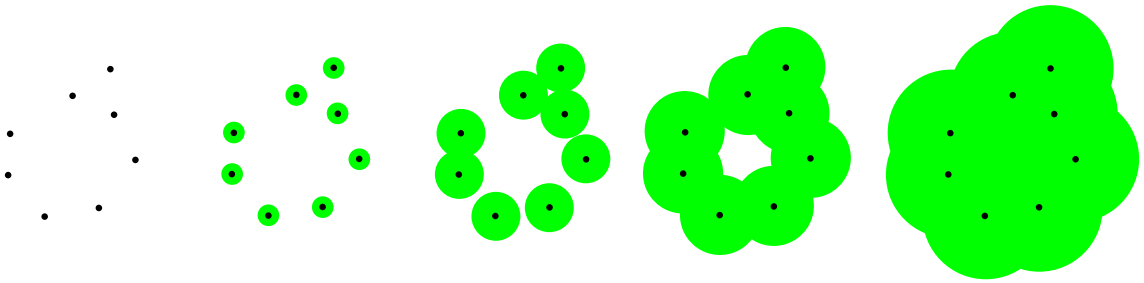


Figure 7.2: As ε increases, the topology of the ε-neighborhoods of P changes.

of persistent homology is to apply the homology functor to the filtration and study the resulting sequence of maps as a whole rather than the homology of each space individually. This sequence of maps not only provides topological information on each space in the filtration but also indicates how the spaces are nested. As a simple example consider the inclusions of a circle in a 2-dimensional torus as on Figure 7.3.

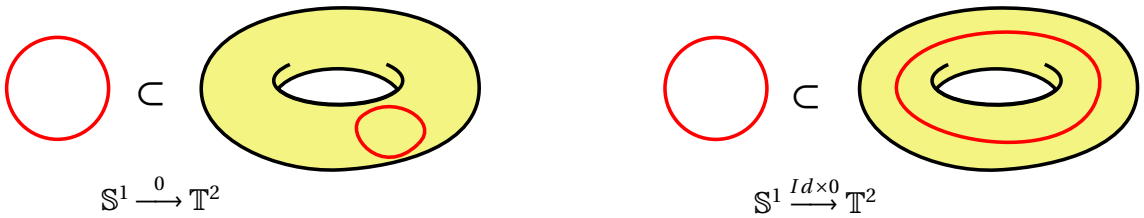


Figure 7.3: The circle may be included as a zero homologous cycle (left) or a non-trivial cycle (right) in the torus. While the induced maps in homology have the same domain and codomain, the maps themselves are distinct.

7.1 Persistence Modules

For computational reasons we shall only consider homology with coefficients in a field $\mathbb{F}$. Hence, a filtration $X_1 \subset X_2 \subset \cdots \subset X_n$ gives rise to a sequence $H_*(X_1) \to H_*(X_2) \to \cdots \to H_*(X_n)$ of linear maps between the vector spaces $H_*(X_i)$. In general, a sequence of linear maps between spaces indexed by an ordered set (typically $[1,n]$, or a subset of $\mathbb{R}$) is called a **persistence module**. The persistence modules over a fixed set of indices form a category. Here, taking $[1,n]$ as indices, a morphism between persistence modules

$(f_i : E_i \to E_{i+1})_{1\leq i<n}$ and $(g_i : F_i \to F_{i+1})_{1\leq i<n}$ is a sequence of linear maps $\phi_i : E_i \to F_i$ that makes the diagram

$$\begin{array}{lccccccc} (f_i): & E_1 & \xrightarrow{f_1} & E_2 & \xrightarrow{f_2} & \dots & \xrightarrow{f_{n-1}} & E_n \\ & \downarrow{\scriptstyle \phi_1} & & \downarrow{\scriptstyle \phi_2} & & & & \downarrow{\scriptstyle \phi_n} \\ (g_i): & F_1 & \xrightarrow{g_1} & F_2 & \xrightarrow{g_2} & \dots & \xrightarrow{g_{n-1}} & F_n \end{array}$$

commute (*i.e.*, $\phi_{i+1} \circ f_i = g_i \circ \phi_i$). The modules (f_i) and (g_i) are isomorphic if we can choose the ϕ_i to be isomorphisms. The **direct sum** of persistence modules (f_i) and (g_i) is the persistence module

$$(f_i) \oplus (g_i) : E_1 \oplus F_1 \xrightarrow{f_1 \oplus g_1} E_2 \oplus F_2 \xrightarrow{f_2 \oplus g_2} \cdots \xrightarrow{f_{n-1} \oplus g_{n-1}} E_n \oplus F_n,$$

where, as usual, $f_i \oplus g_i$ maps $(x, y) \in E_i \oplus F_i$ to $(f_i(x), g_i(y))$.

7.1.1 Classification of Persistence Modules

A persistence module is **decomposable** if it is isomorphic to the direct sum of two non-trivial persistence modules. It is otherwise **indecomposable**. In this section we only consider finite persistence modules indexed over $[1, n]$, and for $1 \leq a \leq b \leq n+1$ we denote by $\mathbb{I}[a, b[$ the persistence module

$$0 \xrightarrow{1} \cdots \xrightarrow{} 0 \xrightarrow{a} \mathbb{F} \xrightarrow{Id} \cdots \xrightarrow{Id} \mathbb{F} \xrightarrow{b-1} 0 \xrightarrow{b} \cdots \xrightarrow{} 0 \xrightarrow{n} 0$$

whose ith space is the 1-dimensional vector space over $\mathbb{F}$ if $i \in [a, b[$ and 0 otherwise.

Exercise 7.1.1. Show that $\mathbb{I}[a, b[$ is indecomposable.

The main result about the classification of persistence modules is the uniqueness of the decomposition into indecomposables.

Theorem 7.1.2. *Let $(f_i)_{1\leq i<n}$ be a persistence module. There exists a unique multiset I of subintervals of $[1, n+1[$ such that*

$$(f_i)_{1\leq i<n} \cong \bigoplus_{[a,b[\in I} \mathbb{I}[a, b[,$$

where each interval in this sum occurs with its multiplicity in I.

The multiset I is the **barcode** of $(f_i)_{1\leq i<n}$. It is composed of **persistence intervals**.

Corollary 7.1.3. *The barcode is a complete invariant for the isomorphism classes of persistence modules.*

Given a persistence module $E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-1}} E_n$, we let $f_{i,j} : E_i \to E_j$ be the composition of the f_k's between e_i and e_j. Precisely, we set

- $\forall i \in [1, n] : f_{i,i} = Id_{E_i}$

- $\forall 1 \le i < j \le n : f_{i,j} = f_{i+1,j} \circ f_i$ and $f_{j,i} = 0$.

We also denote the rank of $f_{i,j}$ by $\beta_{i,j}$. The multiplicity of interval $[i, j[$ in the barcode I is denoted by $m_{i,j}$.

Lemma 7.1.4. $m_{i,j} = (\beta_{i,j-1} - \beta_{i-1,j-1}) - (\beta_{i,j} - \beta_{i-1,j})$

PROOF. Suppose that $E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-1}} E_n \cong \bigoplus_{[a,b[\in I} \mathbb{I}[a, b[$. We easily compute

$$\beta_{i,j}(\mathbb{I}[a, b[) = \begin{cases} 1 & \text{if } [i, j] \subset [a, b[\\ 0 & \text{else.} \end{cases}$$

Note that for any persistence modules (g_k) and (h_k) we have $\beta_{i,j}((g_k) \oplus (h_k)) = \beta_{i,j}((g_k)) + \beta_{i,j}((h_k))$. It follows that $\beta_{i,j}((f_k))$ counts the number of persistence intervals of (f_k) that contain $[i, j]$. Hence, $\delta_{i,j} := \beta_{i,j} - \beta_{i-1,j}$ counts the number of persistence intervals of the form $[i, \ell[$, $\ell > j$. We infer that $m_{i,j} = \delta_{i,j-1} - \delta_{i,j} = (\beta_{i,j-1} - \beta_{i-1,j-1}) - (\beta_{i,j} - \beta_{i-1,j})$. □

Consider a vector $x \in E_i$ in the persistence module $(f_i)_{1 \le i < n} = E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-1}} E_n$. We put $x(j) := f_{i,j}(x)$. A **compatible basis** is a family of vectors $X \subset \bigcup_i E_i$, the union being considered as disjoint, so that

$$X(i) := \{x(i) \mid (x \in X) \wedge (x(i) \neq 0)\}$$

is a basis of E_i for $1 \le i \le n$. In particular, $x, y \in X$ and $x(i) \neq 0$ implies $y(i) \neq x(i)$. The **persistence interval** of $x \in X$ is defined as $I_x = \{i \mid x(i) \neq 0\}$. For convenience, we introduce the *activation function* $a : \bigcup_i E_i \to [1, n]$ such that $x \in E_{a(x)}$ for all $x \in \bigcup_i E_i$. Hence, the lower bound of I_x is $a(x)$.

Lemma 7.1.5. *If $(f_i)_{1 \le i < n}$ admits a compatible basis X, then $(f_i)_{1 \le i < n}$ has a decomposition whose barcode is the multiset of persistence intervals $\{I_x \mid x \in X\}$.*

PROOF. $\bigoplus_{x \in X} \mathbb{I}(I_x)$ has an obvious compatible basis Y obtained by choosing for every $x \in X$ a generator of $\mathbb{F}$ at index $a(x)$. It remains to check that the persistence modules $(f_i)_i$ and $\bigoplus_{x \in X} \mathbb{I}(I_x)$ are isomorphic by constructing an isomorphism sending the bases $X(i)$ to $Y(i)$. □

Proposition 7.1.6. *Every persistence module admits a compatible basis.*

PROOF. For a persistence module $E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-1}} E_n$ we build a compatible basis by induction on n. If $n = 1$, a compatible basis is provided by any basis of E_1. We next assume to have constructed a compatible basis X for

$$E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-2}} E_{n-1}. \tag{7.1}$$

Let $k = |X|$ be the number of basis vectors in X. We recursively define compatible bases $X_1 = X, X_2, \ldots, X_k$ for (7.1). The goal is to get a compatible basis X_k such that

$\{x(n) \mid x \in X_k \wedge x(n) \neq 0\}$ is an independent family in E_n. To this end we first order the elements $x_1, x_2, \ldots, x_k$ of X in a non-decreasing fashion with respect to activation, *i.e.* such that $1 \leq j < k$ implies $a(x_j) \leq a(x_{j+1})$. Suppose we have constructed $X_{i-1} = \{y_1, y_2, \ldots, y_k\}$ for some $k \geq i > 1$, such that the y_j are indexed in non-decreasing order for activation, and such that the nonzero vectors in $\{y_1(n), y_2(n), \ldots, y_{i-1}(n)\}$ form an independent family in E_n.

- If $y_i(n) = 0$ or if $\{y_1(n), y_2(n), \ldots, y_i(n)\}$ is independent, we set $X_i = X_{i-1}$,
- otherwise, we may write $y_i(n) = \sum_{j<i} \lambda_j y_j(n)$. We then put $y_i' = y_i - \sum_{j<i} \lambda_j y_j(i)$, so that $y_i'(n) = 0$, and set $X_i = X_{i-1} \setminus \{y_i\} \cup \{y_i'\}$.

In both cases it is easily seen that X_i is a compatible basis for (7.1). By construction the nonzero images in E_n of the i first vectors in X_i form an independent family. By induction, X_k satisfies our goal. It remains to complete X_k with a basis of a complementary space of $f_{n-1}(E_{n-1})$ in E_n to obtain a compatible basis for $E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-1}} E_n$. □

PROOF OF THEOREM 7.1.2. By Proposition 7.1.6, the persistence module $(f_i)_i$ has a compatible basis, hence a decomposition into indecomposable modules of the form $\mathbb{I}[a, b[$ by Lemma 7.1.5. This decomposition is determined by its barcode which is uniquely defined according to Lemma 7.1.4. □

7.1.2 Restrictions of Persistence Modules

The barcode of a persistence module and of its sub-sequences can be easily related. This relationship will be used in the proof of the stability theorem in Section 7.4.1. In order to formalize the relation, consider a strictly increasing map $\kappa : [1, m] \to [1, n]$. The restriction to κ of the persistence module $(f_i) : E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{n-1}} E_n$ is the persistence module

$$(f_i)|_\kappa : E_{\kappa(1)} \xrightarrow{f_{\kappa(1),\kappa(2)}} E_{\kappa(2)} \cdots \xrightarrow{f_{\kappa(m-1),\kappa(m)}} E_{\kappa(m)}$$

where $f_{i,j}$ was defined below Corollary 7.1.3. Consider the map

$$\begin{array}{rcl} \mu : [1, n+1] & \to & [1, m+1] \\ i & \mapsto & \min\{j \in [1, m+1] \mid \kappa(j) \geq i\} \end{array}$$

where by convention $\kappa(m+1) = n+1$. It is not difficult to see that

$$\mathbb{I}[a, b[|_\kappa = \begin{cases} \mathbb{I}[\mu(a), \mu(b)[& \text{if } \mu(a) < \mu(b) \\ 0 & \text{otherwise.} \end{cases}$$

As an immediate consequence:

Lemma 7.1.7. *Let I be the barcode of a persistence module. The barcode of its restriction to κ is the multiset $\{[\mu(a), \mu(b)[\}_{[a,b[\in I \text{ and } \mu(a) < \mu(b)}$.*

Exercise 7.1.8. Prove the above evaluation for $\mathbb{I}[a, b[|_\kappa$.

7.2 Application to Topological Inference

As explained in the introduction one of the main motivation for the persistence homology theory is the ability to recover the topology of a shape from a sampled set of points, say P. We further remarked that it is appropriate to study the filtration $(P^\varepsilon)_{\varepsilon\in\mathbb{R}_+}$ of the ε-neighborhoods for ϵ ranging from 0 to infinity. We are thus faced with the computation of the barcode of the corresponding induced persistence module. In general, it is more convenient to use simplicial complexes to represent topological spaces in a computer. In particular, the computation of homology groups becomes relatively easy, as seen in the preceding Chapter 6. In order to reduce the filtration (P^ε) to a filtration of a simplicial complexes, we can rely on the following nerve theorem. The **nerve** of a cover $(U_i)_{i\in I}$ of a space X is the abstract simplicial complex whose set of vertices is I and whose simplices are subsets $J\subset I$ such that $\cap_{j\in J}U_j\neq\emptyset$. See Figure 7.4 for an illustration. A cover $(U_i)_{i\in I}$ is **good** if its parts U_i are open sets and if any nonempty intersection of U_i's is contractible[1].

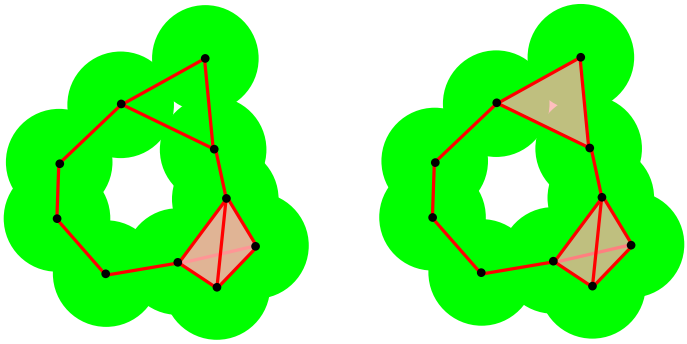


Figure 7.4: Left, the nerve, or Čech complex of a union of balls. Right, the Rips complexe with parameter the diameter of the balls.

Theorem 7.2.1 (Nerve –, Leray'1945, Borsuk'1948)**.** *Let $(U_i)_{i\in I}$ be a good cover of X, then the nerve of $(U_i)_{i\in I}$ has the same homotopy type as X.*

Considering the open balls of radius ε as a cover of P^ε, we observe thanks to the convexity of the balls that they constitute a good cover. Their nerve $C^\varepsilon(P)$ thus has the same homotopy type as P^ε. This nerve is sometimes called the **Čech complex** of P of parameter ε. What is more, if $\varepsilon<\eta$ the inclusions $P^\varepsilon\hookrightarrow P^\eta$ and $C^\varepsilon(P)\hookrightarrow C^\eta(P)$ (as a subcomplex) form a commutative diagram with the homotopy equivalences provided by the nerve theorem $C^\varepsilon(P)\hookleftarrow P^\varepsilon$ and $C^\eta(P)\hookleftarrow P^\eta$:

$$\begin{array}{ccc} C^\varepsilon(P) & \hookrightarrow & C^\eta(P) \\ \updownarrow & & \updownarrow \\ P^\varepsilon & \hookrightarrow & P^\eta \end{array}$$

[1] *i.e.*, has the homotopy type of a point, where two spaces have the same **homotopy type** if there exist maps $f:X\to Y$ and $g:Y\to X$, called **homotopy equivalences**, such that $g\circ f$ is homotopic to the identity on X and $f\circ g$ is homotopic to the identity on Y.

See [CO08] for a proof. It follows that the persistence modules induced by the filtrations $(P^\varepsilon)_{\varepsilon\in\mathbb{R}_+}$ and $(C^\varepsilon(P))_{\varepsilon\in\mathbb{R}_+}$ are isomorphic, hence have the same barcode.

In practice, the construction of $C^\varepsilon(P)$ from P and ε is not very efficient. One should check for every subset of P if the corresponding ε-balls have a common intersection. This is why the Rips complex is sometimes preferred. The **Rips complex** $R^\varepsilon(P)$ of parameter ε associated with P is the **clique complex** of the graph over P where two points are linked by an edge if they are at distance less than ϵ. Recall that the clique complex of a graph is a simplicial complex over the vertices of the graph and has a simplex for every clique of the graph. Hence, event though the Rips complexe can be bigger than the Čech complex it is much easier to compute and can be concisely encoded by its graph. Furthermore, it is easily seen that

$$C^{\varepsilon/2}(P)\subset R^\varepsilon(P)\subset C^\varepsilon(P)$$

In fact, for $P\subset\mathbb{R}^d$ it was shown [DSG07] that

$$R^\varepsilon(P)\subset C^{\eta/2}(P)\subset R^\eta(P)$$

with $\eta=\varepsilon\sqrt{\frac{2d}{d+1}}$. Such relations can be used to replace the Čech complex by the Rips complex in the computations of the barcode. See [DSG07] for more details. The next section explains how to compute the barcode of a filtration of simplicial complexes.

7.3 Computing the Barcode

Consider a filtration $\mathscr{K}: K_1\subset K_2\subset\ldots\subset K_n$ of a simplicial complex $K=K_n$. We want to compute the barcode $I(\mathscr{K})$ of the induced persistence module. In practice we restrict to **simple filtrations** for which each $K_i=K_{i-1}\cup\sigma_i$ is obtained by adding a single simplex σ_i to K_{i-1} (by convention $K_0=\emptyset$). Thanks to Lemma 7.1.7, this actually allows to compute the barcode of non-simple filtrations.

We fix a coefficient field $\mathbb{F}$ and denote by $C(K_i), Z(K_i)$ and $B(K_i)$ the $\mathbb{F}$-vector spaces of chains, cycles and boundaries of K_i, respectively. Hence, the homology group of K_i (actually an $\mathbb{F}$-vector space) is given by $H(K_i)=\ker\partial/\mathrm{Im}\,\partial=Z(K_i)/B(K_i)$, where $\partial: C(K_i)\to C(K_i)$ is the boundary operator. We omit the dimension of the relevant simplices in $C(K_i), Z(K_i), B(K_i)$ and $H(K_i)$, considering that $C(K_i)$ (resp. $Z(K_i)\ldots$) is the direct sum of the chain spaces $C_k(K_i)$ for each dimension k. By the rank-nullity theorem applied to the boundary operator:

$$\dim C(K_i)=\dim Z(K_i)+\dim B(K_i)$$

Noting that $\dim C(K_i)$ is the number of simplices in K_i, we get

$$\big(\dim Z(K_i)-\dim Z(K_{i-1})\big)+\big(\dim B(K_i)-\dim B(K_{i-1})\big)=1$$

Since $\dim Z(K_i)\geq Z(K_{i-1})$ and $\dim B(K_i)\geq B(K_{i-1})$, we have

1. either $\dim Z(K_i)=\dim Z(K_{i-1})+1$ and $B(K_i)=B(K_{i-1})$,
2. or $Z(K_i)=Z(K_{i-1})$ and $\dim B(K_i)=\dim B(K_{i-1})+1$.

We say that index i (or simplex σ_i) is **positive** in the first case and **negative** in the other case. We denote by $\mathscr{P}(\mathscr{K})$ and $\mathscr{N}(\mathscr{K})$ the set of positive, respectively negative, indices.

Lemma 7.3.1. *The following are equivalent:*

- σ_i *is positive,*
- σ_i *is in the support of a cycle* $z \in Z(K_i)$*. Moreover,* $Z(K_i) = Z(K_{i-1}) \oplus \mathbb{F}z$*,*
- $\partial\sigma_i \in B(K_{i-1})$*,*

The proof is left as an exercise. See Figure 7.5. Note that in any case,

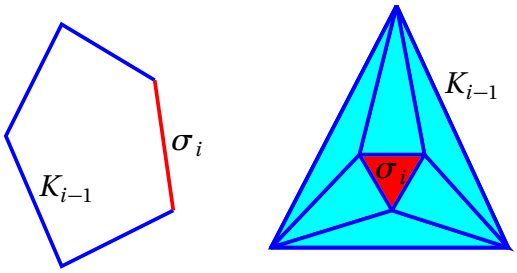


Figure 7.5: Left, σ_i belongs to a cycle of K_i and is thus positive. Right, σ_i is negative.

$$B(K_i) = B(K_{i-1}) + \mathbb{F}\partial\sigma_i. \tag{7.2}$$

The above sum is direct if and only if σ_i est negative. The endpoints a and b of the persistence interval $[a,b[$ are respectively called its lower and upper bound.

Lemma 7.3.2. *Every persistence interval* $[i,j[\in I(\mathscr{K})$ *satisfies*

$$(i,j) \in \mathscr{P}(\mathscr{K}) \times (\mathscr{N}(\mathscr{K}) \cup \{n+1\}).$$

Moreover,

- *Each positive index is the lower bound of a unique interval in* $I(\mathscr{K})$*.*
- *Each negative index is the upper bound of a unique interval in* $I(\mathscr{K})$*.*

Note that $n+1$ is not an index of the filtration and that it may be the upper bound of several persistence intervals.

PROOF. The morphism $\varphi_{i-1} : H(K_{i-1}) \to H(K_i)$ is a quotient of the inclusion $Z(K_{i-1}) \subset Z(K_i)$ by $B(K_{i-1})$ at the domain and by $B(K_i)$ at the codomain. From the definition of a positive simplex it follows that φ_{i-1} is one-to-one and that $\dim H(K_i) = \dim H(K_{i-1}) + 1$ when σ_i is positive. In this case i cannot be the upper bound of a persistence interval of the form $[a,i[$. Indeed, the corresponding indecomposable module $\mathbb{I}[a,i[$ would appear in the decomposition of $(H(K_i))_i$. However, the segment of $\mathbb{I}[a,i[$ between index $i-1$ and i is the map $\mathbb{F} \to 0$, which is obviously not injective. Similarly, if σ_i

is negative then φ_{i-1} is onto and $\dim H(K_i) = \dim H(K_{i-1}) - 1$. As a consequence, i cannot be the lower bound of any persistence interval. On the other hand, $\dim H(K_i)$ is the number of persistence intervals that contain i. It easily follows that exactly one interval starts when σ_i is positive and one interval ends when σ_i is negative. □

We can thus define the **birth function** as the map $b : \mathcal{N}(\mathcal{K}) \to \mathcal{P}(\mathcal{K})$ such that for all $j \in \mathcal{N}(\mathcal{K})$, $[b(j), j[\in I(\mathcal{K})$. In particular,

$$I(\mathcal{K}) = \{[b(j), j[\}_{j \in \mathcal{N}(\mathcal{K})} \cup \{[i, n+1[\}_{i \in \mathcal{P}(\mathcal{K}) \setminus \operatorname{Im} b} \tag{7.3}$$

Hence, we may recover the barcode $I(\mathcal{K})$ from the knowledge of the signs of the simplices and of the birth function.

7.3.1 Compatible Boundary Basis

A **Compatible boundary basis** is a family of cycles $\mathcal{B}(\mathcal{K}) = \{x_j\}_{j \in J} \subset Z(K)$, with $J \subset [1, n]$, such that:

1. $\forall i \in [1, n], \{x_j\}_{j \in J \cap [1,i]}$ is a basis of $B(K_i)$,
2. the map $\beta : J \to [1, n]$, $j \mapsto$ (maximum index of the simplices in x_j) is injective.

Lemma 7.3.3. *Suppose that $\mathcal{K}$ has a compatible boundary basis, then β coincides with the birth function b.*

PROOF. The above Condition 1 and the remark after Equation (7.2) show that $J = \mathcal{N}(\mathcal{K})$. Lemma 7.3.1 also implies that $\beta(j) \in \mathcal{P}(\mathcal{K})$ for all $j \in J$. For every $i \in \mathcal{P}(\mathcal{K})$, define $z_i \in Z(K_i)$ as follows.

- If $i = \beta(j)$ for some $j \in J$, then $z_i = x_j$.
- Else, choose z_i such that $Z(K_i) = Z(K_{i-1}) \oplus \mathbb{F} z_i$ (cf. Lemma 7.3.1).

Remark that the simplex with maximum index in the support of z_i is σ_i. Hence, (cf. Lemma 7.3.1) $\{z_j\}_{j \in \mathcal{P}(\mathcal{K}), j \le i}$ is a basis of $Z(K_i)$. Let $[z]_i$ denote the homology class of cycle z in $H(K_i)$. We need to check that $([z_j]_j)_{j \in \mathcal{P}(\mathcal{K})}$ is a compatible basis for the homology sequence of $\mathcal{K}$ and that the persistence interval of each $[z_{\beta(j)}]_{\beta(j)}$ is $[\beta(j), j[$, while the persistence interval of $[z_j]_j$, $j \in \mathcal{P}(\mathcal{K}) \setminus \beta(J)$, is $[j, n+1[$. We claim that

$$Z(i) := \{[z_{\beta(j)}]_i\}_{(j \in J) \wedge (\beta(j) \le i) \wedge (j > i)} \cup \{[z_j]_i\}_{(j \le i) \wedge (j \in \mathcal{P}(\mathcal{K}) \setminus \beta(J))}$$

is a basis of $H(K_i)$. Since $[z_{\beta(j)}]_j = [x_j]_j = 0$, we also have $[z_{\beta(j)}]_i = 0$ for $i \ge j$ and it follows from the above remark that $Z(i)$ spans $H(K_i)$. To see that $Z(i)$ is an independent set, consider a linear combination $\sum_{(j \in J) \wedge (\beta(j) \le i) \wedge (j > i)} \alpha_j [z_{\beta(j)}]_i + \sum_{(j \le i) \wedge (j \in \mathcal{P}(\mathcal{K}) \setminus \beta(J))} \alpha_j [z_j]_i$ of elements in $Z(i)$. If it is zero, then the combination $c := \sum_{(j \in J) \wedge (\beta(j) \le i) \wedge (j > i)} \alpha_j z_{\beta(j)} + \sum_{(j \le i) \wedge (j \in \mathcal{P}(\mathcal{K}) \setminus \beta(J))} \alpha_j z_j$ of the corresponding cycles must lie in $B(K_i)$. By the first condition in the definition of a compatible boundary basis, cycle c must be equal to a linear combination of $\{x_j \mid (j \in J) \wedge (j \le i)\}$. Because the maximum index of the simplices in the support of each $z_{\beta(j)}, z_j$ and x_j are pairwise distinct, it must be that all the coefficients α_j in c are null, thus concluding the proof of the claim. We finally observe that the persistence interval of $[z_j]_j$ is the set of i's for which $[z_j]_i \in Z(i)$. Whence, for $j \in J$ the persistence interval of $[z_{\beta(j)}]_{\beta(j)}$ is $[\beta(j), j[$, while for $j \in \mathcal{P}(\mathcal{K}) \setminus \beta(J)$ the persistence interval of $[z_j]_j$ is $[j, n+1[$. □

7.3.2 Algorithm

Lemma 7.3.3 and Equation (7.3) show that it is enough to construct a compatible boundary basis for $\mathscr{K}$ to derive the sign of each simplex and the barcode of $\mathscr{K}$. We can construct a compatible boundary basis by induction on the size n of the filtration. The base case $n = 1$ is trivial because the unique simplex in the filtration must be a (positive) vertex. We thus assume that we have computed a compatible boundary basis $\mathscr{B}(\mathscr{K}') = \{x_j\}_{j\in J}$ for the sub-filtration $\mathscr{K}'$:

$$K_1 \subset K_2 \subset \ldots \subset K_{n-1}$$

We denote by $b : J \to \mathscr{P}(\mathscr{K}')$ the corresponding birth function. Suppose that we can write

$$\partial \sigma_n = \sum_{j\in J} \alpha_j x_j + y, \tag{7.4}$$

where

1. either $y = 0$,
2. or the maximum index of the simplices in y is not in $b(J)$.

In case 1, we have $B(K_n) = B(K_{n-1})$ and $\mathscr{B}(\mathscr{K}')$ remains a compatible boundary basis for $\mathscr{K}$. In case 2, n is negative and $\mathscr{B}(\mathscr{K}') \cup \{y\}$ is a compatible boundary basis for $\mathscr{K}$.

By the second condition in its definition, every compatible boundary basis is in echelon form when the cycles are written as combination of simplices. We can thus apply Gaussian elimination as in the following pseudocode to obtain a decomposition as in (7.4).

$y := \partial \sigma_n$
$i :=$ maximum index of the simplices in y
While $((y \neq 0) \wedge (i \in b(J)))$
  $j := b^{-1}(i)$
  $\alpha :=$ coefficient of σ_i in y
  $\beta :=$ coefficient of σ_i in x_j
  $y := y - (\alpha/\beta)x_j$
  $i :=$ maximum index of the simplices in y * *undefined if* $y = 0$ **\
End while
* $y = 0$ *or* $y = x_n$ *when leaving the while loop* *\

We can store each x_j as a table of coefficients indexed by the n simplices of the filtration. We represent the birth function as a table of length n; the jth entry contains $b(j)$ if j is negative and 0 otherwise. We also store the inverse map b^{-1} in a table of length n. The computation of x_n by the above loop takes $O(n^2)$ time. Hence,

Proposition 7.3.4. *We can compute a compatible boundary basis and the birth function of a filtration of length n in $O(n^3)$ time on an $\mathbb{F}$-RAM machine. We can moreover compute the barcode of the filtration in the same amount of time.*

7.4 Persistence Diagrams

A function $f : K \to \mathbb{R}$ over a simplicial complex K is **non-decreasing** if

$$\forall \sigma, \tau \in K : \sigma \prec \tau \implies f(\sigma) \leq f(\tau)$$

where $\sigma \prec \tau$ means "σ *is a face of* τ". A filtration of K can be equivalently described by a non-decreasing function f over K. Indeed, if $f_1 < f_2 < \ldots < f_n$ is the sequence of *values* of f, then the sequence

$$f^{-1}([-\infty, f_1]) \subset f^{-1}([-\infty, f_2]) \subset \ldots \subset f^{-1}([-\infty, f_n]) \tag{7.5}$$

is a filtration of K, which we denote by $\mathscr{K}_f$. Conversely, any filtration $K_1 \subset K_2 \subset \ldots \subset K_n = K$ has the form $\mathscr{K}_f$ for f defined over K by $f(\sigma) = i \Leftrightarrow \sigma \in K_i \setminus K_{i-1}$.

We set $f_{n+1} = +\infty$. The **persistence diagram** $D(f)$ of f is the multiset of points in the extended plane $(\mathbb{R} \cup \{-\infty, +\infty\})^2$ given by

$$D(f) = \{(f_i, f_j)\}_{[i,j[\in I(\mathscr{K}_f)} \cup \Delta^\infty,$$

where Δ^∞ is the multiset of points on the diagonal $\{x = y\}$, each counted with countably infinite multiplicity. We say that the filtration $\mathscr{K} : K_1 \subset K_2 \subset \ldots \subset K_m = K$ is **compatible** with $f : K \to \mathbb{R}$ if $\mathscr{K}_f$ is a sub-filtration of $\mathscr{K}$. In other words, $\mathscr{K}$ is compatible with f if f is constant over each $K_i \setminus K_{i-1}$ and if $f_{\mathscr{K}} : [1, m] \to \mathbb{R}$, $i \mapsto f(K_i \setminus K_{i-1})$ is non-decreasing. In this case we define the **persistence diagram of f relatively to $\mathscr{K}$** as the multiset:

$$D(f, \mathscr{K}) = \{\big(f_{\mathscr{K}}(i), f_{\mathscr{K}}(j)\big)\}_{[i,j[\in I(\mathscr{K})} \cup \Delta^\infty$$

where we have put $f_{\mathscr{K}}(m+1) = +\infty$. In particular, $D(f) = D(f, \mathscr{K}_f)$.

Lemma 7.4.1. *$D(f) = D(f, \mathscr{K})$ for any filtration $\mathscr{K}$ compatible with f.*

PROOF. Let $f_1 < f_2 < \ldots < f_n$ be the sequence of values of f over K. We set

$$\kappa : [1, n] \to [1, m], i \mapsto \max\{j \mid f_{\mathscr{K}}(j) = f_i\}.$$

Hence, $f^{-1}([-\infty, f_i]) = K_{\kappa(i)}$ and the persistence module induced by the homology of $\mathscr{K}_f$ is the restriction to κ of the persistence module induced by $\mathscr{K}$ (see Section 7.1.2). By Lemma 7.1.7 we have $I(\mathscr{K}_f) = \big\{[\mu(i), \mu(j)[\big\}_{[i,j[\in I(\mathscr{K}) \text{ and } \mu(i) < \mu(j)}$, with μ as in Lemma 7.1.7. It follows that

$$D(f) = D(f, \mathscr{K}_f) = \{\big(f_{\mu(i)}, f_{\mu(j)}\big) \mid [i, j[\in I(\mathscr{K}) \text{ and } \mu(i) < \mu(j)\} \cup \Delta^\infty$$

We easily check from the definitions of κ and μ that $f_{\mu(i)} = f_{\mathscr{K}}(i)$. Hence,

$$D(f) = \{\big(f_{\mathscr{K}}(i), f_{\mathscr{K}}(j)\big) \mid [i, j[\in I(\mathscr{K}) \text{ and } \mu(i) < \mu(j)\} \cup \Delta^\infty$$

Now, if $\mu(i) \geq \mu(j)$ for some interval $[i, j[\in I(\mathscr{K})$ then $f_{\mathscr{K}}(i) = f_{\mathscr{K}}(j)$ and the corresponding point $(f_{\mathscr{K}}(i), f_{\mathscr{K}}(i))$ is "absorbed" by the diagonal Δ^∞. We finally conclude $D(f) = \Delta^\infty \cup \{\big(f_{\mathscr{K}}(i), f_{\mathscr{K}}(j)\big)\}_{[i,j[\in I(\mathscr{K})} = D(f, \mathscr{K})$. □

7.4.1 Stability of Persistence Diagrams

The stability of the persistence diagram $D(f)$ with respect to f is the main result of Persistence theory. We first introduce the **bottleneck distance** d_B between persistence diagrams. Note that thanks to the diagonal Δ^∞ any two diagrams D, D' are in bijection. We set

$$d_B(D,D') = \inf_{\phi} \sup_{p\in D} \|p-\phi(p)\|_\infty$$

where $\phi : D \to D'$ runs over the bijections between D and D' and $\|p-q\|_\infty = \max\{|x_p - x_q|, |y_p - y_q|\}$ (by convention, $|+\infty - x| = 0$ if $x = +\infty$ and $|+\infty - x| = +\infty$ otherwise). See Figure 7.6. Note that d_B is not a distance properly speaking: it can take infinite

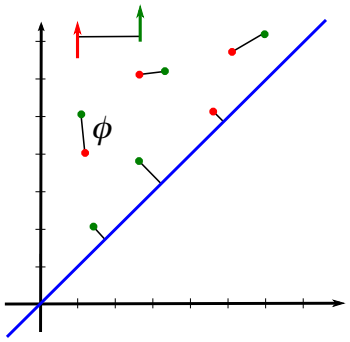


Figure 7.6: The bottleneck distance is computed by minimizing over all bijections ϕ the largest distance in each pairing.

values but otherwise satisfies the triangular inequality.

As usual, for any functions $f, g : K \to \mathbb{R}$, we denote their L_∞ distance by

$$\|f-g\|_\infty = \sup_{\sigma\in K} |f(\sigma) - g(\sigma)|$$

Theorem 7.4.2 (Stability –, [CSEH07, CSEM06]). $d_B(D(f), D(g)) \le \|f-g\|_\infty$

PROOF. Put $f_t = f + t(g-f)$. note that if f, g are non-decreasing over K, so is f_t. For every two simplices $\sigma, \tau \in K$, there exists $u \in [0,1]$ such that the sign of $f_t(\sigma) - f_t(\tau)$ is constant for $t \in [0,u]$ and the same is true for $t \in [u,1]$. There is thus a finite partition $0 = t_0 < t_1 < \ldots < t_r = 1$ of[2] $[0,1]$ so that the relative order of the f_t-values of the simplices is independent of t over each interval $[t_i, t_{i+1}]$. It follows that for each $i \in [0, r-1]$ we can exhibit a simple filtration $\mathscr{K}_i$ compatible with *every* function f_t for $t \in [t_i, t_{i+1}]$. By Lemma 7.4.1, we have

$$D(f_t) = D(f_t, \mathscr{K}_i) = \Delta^\infty \cup \{\big(f_t(\sigma_a), f_t(\sigma_b)\big)\}_{[a,b[\in I(\mathscr{K}_i)}$$

where σ_a is the ath simplex of $\mathscr{K}_i$. Considering the obvious correspondence between $D(f_{t_i})$ and $D(f_{t_{i+1}})$ that restricts to the identity over Δ^∞ and sends $\big(f_{t_i}(\sigma_a), f_{t_i}(\sigma_b)\big)$ to $\big(f_{t_{i+1}}(\sigma_a), f_{t_{i+1}}(\sigma_b)\big)$, we infer $d_B(D(f_{t_i}), D(f_{t_{i+1}})) \le (t_{i+1} - t_i)\|f-g\|_\infty$. Applying the triangular inequality we finally conclude

$$d_B(D(f), D(g)) \le \sum_i (t_{i+1} - t_i)\|f-g\|_\infty = \|f-g\|_\infty$$

[2] $r \le \binom{m}{2} + 1$ where m is the number of simplices of K

□

The Stability theorem was refined in a more general context by Chazal et al. and Bubenik and Scott [CCSG+09, CDSGO12, BS14]. A first generalization is to consider "continuous" persistence module indexed over $\mathbb{R}$. This is a family of vector spaces $(V_x)_{x\in\mathbb{R}}$ and a family of linear maps $(v_{x,y} : V_x \to V_y)_{x\leq y}$ satisfying $v_{x,x} = Id_{V_x}$ and $v_{x,z} = v_{x,y} \circ v_{y,z}$ for all $x \leq y \leq z$. We denote it by $\mathbb{V}$. Given two persistence modules $\mathbb{V}$ and $\mathbb{W}$ over $\mathbb{R}$ and a real number d, a **degree d morphism** $\varphi : \mathbb{V} \to \mathbb{W}$ is a family of linear maps $(\varphi_x : V_x \to W_{x+d})_{x\in\mathbb{R}}$ such that the following diagram:

$$\begin{array}{ccccc} V_x & \longrightarrow & V_y & & \\ & \searrow^{\varphi_x} & & \searrow^{\varphi_y} & \\ & & W_{x+\varepsilon} & \to & W_{y+\varepsilon} \end{array}$$

commutes for all $x \in \mathbb{R}$. An **ε-interleaving** is a pair of morphisms $\varphi : \mathbb{V} \to \mathbb{W}$ and $\psi : \mathbb{W} \to \mathbb{V}$, each of degree ε, such that the following diagrams:

$$\begin{array}{ccc} V_x & \longrightarrow & V_{x+2\varepsilon} \\ & \searrow^{\varphi_x} \quad \nearrow_{\psi_{x+\varepsilon}} & \\ & W_{x+\varepsilon} & \end{array} \qquad \begin{array}{ccc} & V_{x+\varepsilon} & \\ \nearrow^{\psi_x} & & \searrow^{\varphi_{x+\varepsilon}} \\ W_x & \longrightarrow & W_{x+2\varepsilon} \end{array}$$

commute. The **interleaving distance** between $\mathbb{V}$, $\mathbb{W}$ is

$$d_i(\mathbb{V},\mathbb{W}) = \inf\{\varepsilon \mid \exists \varepsilon\text{-interleaving between } \mathbb{V},\mathbb{W}\}$$

When is $\mathbb{V}$ is pointwise finite dimensional, *i.e.*, when each space V_x if finite dimensional, it can be shown that the decomposition Theorem 7.1.2 remains valid [CB15]. This time each indecomposable module $\mathbb{I}(\iota)$ may apply to any type of interval ι (half-open, closed, semi-infinite,...) and satisfies $\mathbb{I}(\iota)_x = \mathbb{F}$ for $x \in \iota$ and $\mathbb{I}(\iota)_x = 0$ otherwise, with identity or zero maps wherever it applies. The persistence diagram is then defined as the multiset of points (u, v) where u, v runs over the endpoints of the persistence intervals.

Theorem 7.4.3 (Isometry –, [CCSG+09, CDSGO12])**.** *Let $\mathbb{V}$ and $\mathbb{W}$ be pointwise finite dimensional persistence modules over $\mathbb{R}$ such that rank $v_{x,y}$ and rank $w_{x,y}$ is finite for every $x < y$. Then,*

$$d_B(D(\mathbb{V}), D(\mathbb{W})) = d_i(\mathbb{V},\mathbb{W})$$

The stability theorem can be deduced from the isometry theorem as follows. Let X be a topological space and $f : X \to \mathbb{R}$. Put $X_t^f := f^{-1}(-\infty, t]$. The filtration $\mathbb{X}^f := (X_t^f)_{t\in\mathbb{R}}$ induces a persistence module $H(\mathbb{X}^f)$ by applying the homology functor. Now, $\|f - g\|_\infty \leq \varepsilon$ implies $X_t^f \subset X_{t+\varepsilon}^g \subset X_{t+2\varepsilon}^f$. It follows that $H(\mathbb{X}^f), H(\mathbb{X}^g)$ are ε-interleaved, whence by the isometry Theorem $d_i(H(\mathbb{X}^f), H(\mathbb{X}^g)) \leq \varepsilon$. In turn, this implies $d_B(D(f), D(g)) = d_i(H_*(X_f), H_*(X_g)) \leq \|f - g\|_\infty$.

Exercise 7.4.4. Check that a persistence module over an ordered set $(X, \leq)$, e.g. $X = \mathbb{R}$, is the same as a functor from the category $(X, \leq)$ to the category of $\mathbb{F}$-vector spaces. Here, the objects of $(X, \leq)$ are the elements of X and each pair (x, y) of objects has exactly one morphism if $x \leq y$ (and none otherwise).

8

Knots and 3-Dimensional Computational Topology

Contents

In the next two courses, we switch our attention from 2-dimensional space (surfaces) to 3-dimensional space, focusing on *knots*, and incidentally dealing a bit with 3-manifolds. This increase in dimension has a dramatic effect from the point of view of computational topology: while most topological problems on surfaces can be solved efficiently, their generalizations in dimension 3 are much harder to understand. As an illustration, while recognizing closed surfaces can be done in linear time by just computing the Euler characteristic and orientability, all the algorithms known to detect whether two 3-dimensional spaces are homeomorphic [Jac05, Kup15a] are very inefficient (requiring more than towers of exponentials) and complicated, relying on

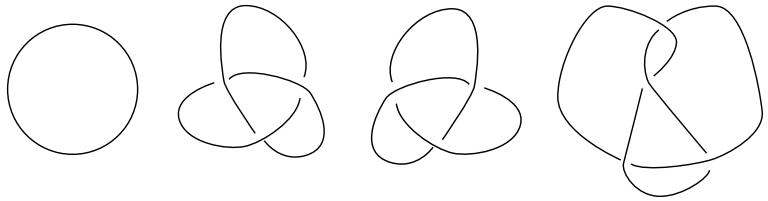


Figure 8.1: Example of knots: the trivial knot, the left and right trefoil knots, and the figure-eight knot. These are not polygonal but can obviously be made so.

Perelman's recent proof [Per02, Per03] of Thurston's Geometrization Conjecture. Nevertheless, despite being hard, most problems in 3 dimensions are decidable, and thus form an interesting middle ground before higher dimensions where indecidability results start to kick in.

8.1 Knots

A **knot** is a closed curve in $\mathbb{R}^3$, or more formally an embedding $\mathbb{S}^1 \to \mathbb{R}^3$. In contrast with the two-dimensional case, allowing arbitrary topological knots might lead to pathological objects known as **wild knots** which, while interesting in their own right, are not very relevant from an algorithmic perspective. So we restrict our attention to **tame knots**, which are polygonal[1] embeddings $\mathbb{S}^1 \to \mathbb{R}^3$. We will omit the word tame throughout these notes. Two knots are considered equivalent if they can be deformed continuously one onto the other one without introducing self-crossings in the process. More formally, the notion of equivalence considered here is through **ambient isotopy**, that is a continuous family of homeomorphisms $h_t : \mathbb{R}^3 \times [0,1] \to \mathbb{R}^3$. Two knots K_1 and K_2 are (ambient) **isotopic** if there exists an ambient isotopy h_t such that $h_0 = Id_{\mathbb{R}^3}$ and $h_1(K_1) = K_2$. Figure 8.1 shows some examples of knots.

Remark 1: By the *Alexander trick* (see for example [BZ85, Proposition 1.9]), any orientation-preserving homeomorphism of the 3-ball $\mathbb{B}^3 \to \mathbb{B}^3$ is actually an ambient isotopy, so homeomorphisms could simply be used to define equivalence, but the notion of ambient isotopies is more intuitive.

Remark 2: On the other hand, our notion of ambient isotopy is a bit unnatural because it breaks the polygonal structure of tame knots. The underlying idea is that we restrict our attention to tame knots to avoid pathologies, but once this restriction has been made, it is much more convenient to allow any kind of continuous deformation. Yet if one insists on only allowing polygonal deformations, one can restrict the allowed isotopies to *elementary moves* or Δ-moves which are illustrated in Figure 8.2. It turns out that knots are equivalent under ambient isotopies if and only if they are equivalent under elementary moves, see [BZ85, Proposition 1.10].

Remark 3: Another tempting definition of equivalence could be to use **isotopies**, i.e., to say that two knots K_1 and K_2 are isotopic if there is a continuous family of embeddings $i_t : \mathbb{S}^1 \to \mathbb{S}^3$ such that $i_0 = K_1$ and $i_1 = K_2$. However, with such a definition

[1] Considering *smooth* knots leads to the same theory.

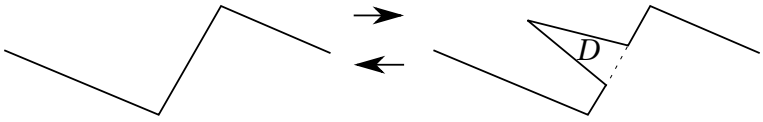


Figure 8.2: An elementary move on a knot: a segment is subdivided and can be moved along a triangle if the disk D does not intersect the rest of the knot.

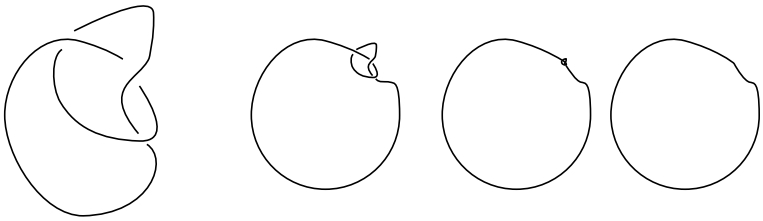


Figure 8.3: Equivalence of all knots using just isotopy.

all the knots would be equivalent, as illustrated in Figure 8.3, since it allows to pull the knotted portion progressively tighter until it disappears.

A **trivial knot** (or unknot) is a knot isotopic to the trivial embedding of the circle $\mathbb{S}^1 \to \mathbb{R}^3$. Since we restrict our attention to polygonal knots, the following problem is a well-defined algorithmic problem, which will be the focus of these two lectures.

> UNKNOT RECOGNITION
> **Input:** A knot K described as a concatenation of n segments.
> **Output:** Is K a trivial knot?

As we will see, this is a tricky problem, the current state of the art is that it lies in **NP** ∩ **co-NP** (see [HLP99, Lac16]), yet no polynomial time algorithm is known for this problem. But before delving into this, let us first observe that it is not even easy to prove that there exist non-trivial knots. In order to prove this, we introduce knot diagrams.

8.2 Knot diagrams

A convenient way to deal with knots is to represent them using **knot diagrams**, that is, a 2-dimensional orthogonal projection $p : \mathbb{R}^3 \to \mathbb{R}^2$ to project K on $\mathbb{R}^2$. We call such a projection regular if there are no triple (or worse) points and no vertex of K is a double point. A knot diagram is obtained from a regular projection by specifying at each crossing which strand is above the other one. By perturbing slightly either a knot or the desired projection if needed, it is easy to associate a knot diagram with any knot. Basically, this is what we did in Figure 8.1 without even bothering to mention it. From the point of view of graph theory, a knot diagram is a 4-regular planar graph where each vertex bears a *marking*, indicating which strand is above the other one.

The **Reidemeister moves** are the local moves relating knot diagrams represented in Figure 8.4. Two knot diagrams are considered equivalent if they can be related with isotopies of $\mathbb{R}^2$ and Reidemeister moves.

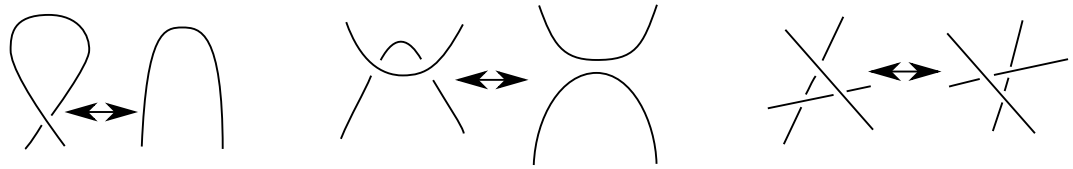


Figure 8.4: The three Reidemeister moves.

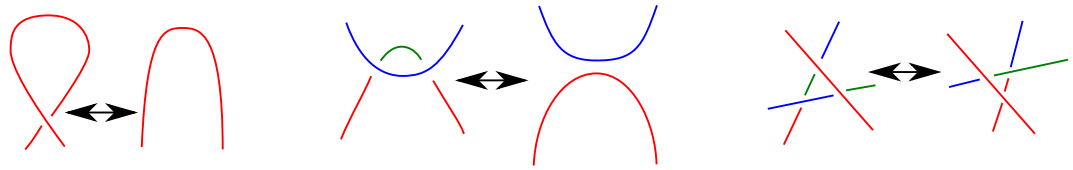


Figure 8.5: Tricoloring the Reidemeister moves.

Theorem 8.2.1 (Reidemeister [Rei27])**.** *Two knots are equivalent if and only if all their diagrams are equivalent.*

PROOF. We first prove that any two regular projections of the same knot are connected by Reidemeister moves. Each projection can be identified with a point on the sphere $\mathbb{S}^2$, and the non-regular projections are correspond to curves on this sphere. By connecting p_1 and p_2 by a path in general position with respect to these lines, it is enough to show that crossing a line of non-regular projections can be done with Reidemeister moves. The three possible situations of the knot around these lines correspond to the three Reidemeister moves. Now, since two equivalent knots can be related using elementary moves (see Remark 2 above), it is enough to show that the projection of an elementary move can be realized with Reidemeister moves, which is easily verified. □

We can leverage on this combinatorial approach to knot equivalence to provide an easy proof that the trefoil knot is non-trivial.

Proposition 8.2.2. *There exists a non-trivial knot.*

PROOF. A knot diagram is said to be **tricolorable** if each strand can be colored using one of three colors, with the following rule:

1. At least two colors must be used.
2. At a crossing, the three incident strands[2] are either all of the same color or all of different colors.

We claim that the tricolorability of a knot does not depend on the knot diagram. By Theorem 8.2.1, it suffices to prove that Reidemeister moves preserve tricolorability, which is illustrated in Figure 8.5.

Now, observing that the trivial knot is not tricolorable (since it can only be colored with one color), while the trefoil knot is (see Figure 8.6), this proves that the trefoil knot is non-trivial. □

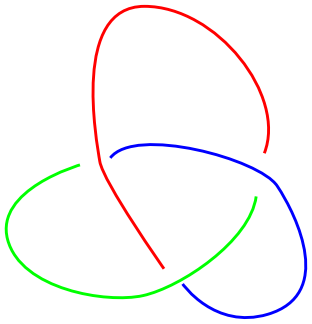


Figure 8.6: Tricoloring the trefoil knot.

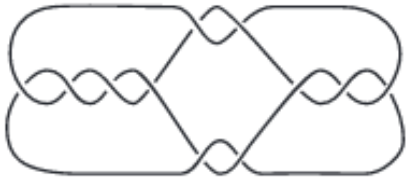


Figure 8.7: Goeritz's unknot.

Note that this does not detect all the non-trivial knots, since the figure-eight knot can not be colored with three colors either (left as an exercise), and it is not trivial (see Exercise 8.3.3). Also note that for algorithmic purposes, this approach, being based on coloring, is very inefficient.

On the other hand, Theorem 8.2.1 suggests a very candid approach to solve UNKNOT RECOGNITION: simply try combinations of Reidemeister moves until one reaches the **trivial diagram**, i.e., an embedding $\mathbb{S}^1 \to \mathbb{R}^2$. Optimistically, one might hope it is never necessary to make a knot more complicated to untangle it, i.e., maybe the Reidemeister move II increasing the number of crossings is not needed to reach the unknot. This would bound the number of combinations to try and give an exponential algorithm. However, there exist **hard unknots**, that is, knot diagrams of the unknot that require to be made more complicated before reaching the unknot. Figure 8.7 shows one of those, and there are infinite families of these.

That approach is not doomed to fail however, and there exist bounds on the number of Reidemeister moves needed to simplify a trivial knot. In a recent breakthrough, Lackenby obtained the following polynomial bound.

Theorem 8.2.3 (Lackenby [Lac15])**.** *Let D be a diagram of the trivial knot with c crossings. Then there exists a series of $(236c)^{11}$ Reidemeister moves transforming it into the trivial diagram.*

This provides an exponential time algorithm to solve UNKNOT RECOGNITION, and also proves that it is in **NP**: the certificate is the sequence of Reidemeister moves to be applied to reach the trivial diagram. As the constant and exponent might suggest, this

[2]Here, the arc going "above" is considered as a single strand, while the arc going "below" is cut into two strands.

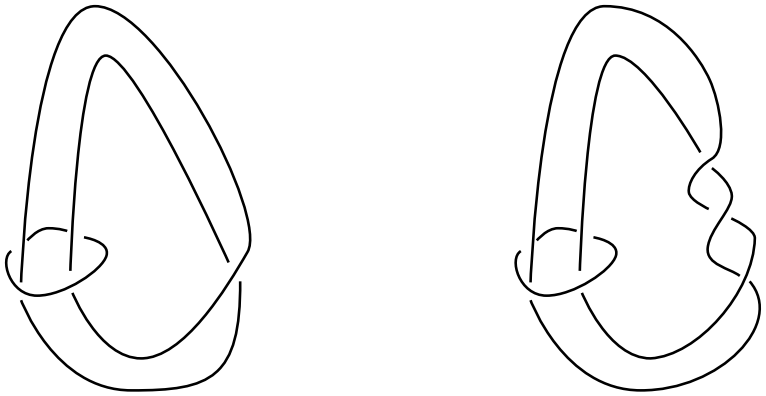


Figure 8.8: Links are not determined by their complements.

theorem is very intricate and we will not prove it in this course. However, we will see another algorithm showing that the problem is in **NP**, based on *normal surface theory*, which is one of the main technical tools in the proof of Theorem 8.2.3. So after this course, you will be better equipped to understand the proof.

8.3 The knot complement

A central way to study knots is to study the topological properties of their complements. For this purpose, it is convenient to compactify $\mathbb{R}^3$, i.e., to add a point and identify the result to $\mathbb{S}^3$ using the stereographic projection. Knots in $\mathbb{R}^3$ and $\mathbb{S}^3$ behave identically, so we will work in this section with the latter framework. For a knot K in $\mathbb{S}^3$, drill a tubular neighborhood N around K and denote the resulting space by $M = \mathbb{S}^3 \setminus N$. This is an example of a **3-manifold with boundary**, i.e., a topological space where every point is locally homeomorphic to $\mathbb{R}^3$ or the half-space $\mathbb{R}^3_{|x\geq 0}$.

The study of knot complements as a tool to understand a knot is justified by the following theorem, which at the same time sounds very obvious and is extremely hard to prove.

Theorem 8.3.1 (Gordon-Luecke [GL89])**.** *Knots are determined by their complements, i.e., if two knots have complements that are homeomorphic with an orientation-preserving homeomorphism, then they are isotopic.*

To emphasize the strength of this theorem, let us just warn the reader that **links**, which are embeddings of disjoint copies of $\mathbb{S}^1$ into $\mathbb{R}^3$, are *not* determined by their complements: Figure 8.8 shows two links which are not ambient isotopic, yet their complements are homeomorphic.

So now that instead of this somewhat strange notion of ambient isotopy, we reduced the problem to determining the homeomorphism class of a space, we can try to apply the tools that we have seen in the previous lectures to solve it. As we shall see, this will not be very fruitful with respects to solving UNKNOT RECOGNITION.

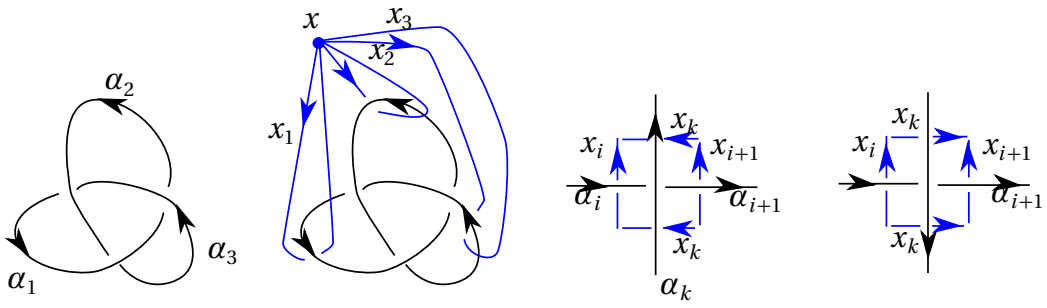


Figure 8.9: The Wirtinger presentation. In the last two pictures, depending on the orientation of the strands, we will have $x_k x_i = x_{i+1} x_k$ or $x_i x_k = x_k x_{i+1}$

8.3.1 Homotopy

One can define the homotopy group π_1 of the topological space $M = \mathbb{S}^3 \setminus N$ in the same way as the topological fundamental group that we introduced for surfaces. We first choose a basepoint x, consider the set of loops based at x and say that two loops are equivalent if they are homotopic (in M, that is, if they can be deformed into each other *without crossing* K). The set of loops obtained this way forms the group $\pi_1(M)$, where the identity element is the trivial loop at x and the law is the concatenation.

It turns out that there is a fairly easy way to obtain a presentation of the group $\pi_1(M)$ via the **Wirtinger presentation**. Start with a knot diagram D of K. As for tricolorability, we consider a *strand* in the diagram to be and arc in the diagram between two points where it goes "below" another arc. Number the strands of D by $\alpha_1, \ldots, \alpha_n$ according to the order in which they appear in D, and orient them according to an arbitrary orientation. Now, let us pick a basepoint x somewhere above the knot (for example in the eye of the reader), and define a set of loops $x_1, \ldots, x_n$ by starting at x and going looping around α_i by passing under it in a right left direction, and going back to x.

Now, at each crossing involving the strands α_i, α_{i+1} and α_k, the loops x_i, x_{i+1} and x_k will verify some relation, which can be simply read depending on the orientation of the crossings, see Figure 8.9. In the first case, we will have $x_k x_i = x_{i+1} x_k$, while in the second case we will have $x_i x_k = x_k x_{i+1}$, we denote by r_i the corresponding relation that holds. It turns out that these relations encapsulate all the possible relations between the generators x_i.

Theorem 8.3.2. *The fundamental group $\pi_1(\mathbb{S}^3 \setminus N)$ admits the presentation*

$$< x_1, \ldots, x_n \mid r_1, \ldots r_n . >$$

PROOF. The formal way to prove this theorem is to decompose $\mathbb{S}^3$ into cells defined by the knot K and apply the *van Kampen theorem*, see Hatcher [Hat02, Section 1.2]. In order to keep things simple, we provide a somewhat vague proof that should be enough to convince the novice reader that the theorem is correct, and to convince the expert reader (who probably already knows all this) that applying the van Kampen theorem really yields this result.

Grow a family of vertical half-planes $Z = Z_1, \ldots Z_n$ under each strand of the knot diagram, as in Figure 8.10. Some of these planes will touch (under a crossing of D).

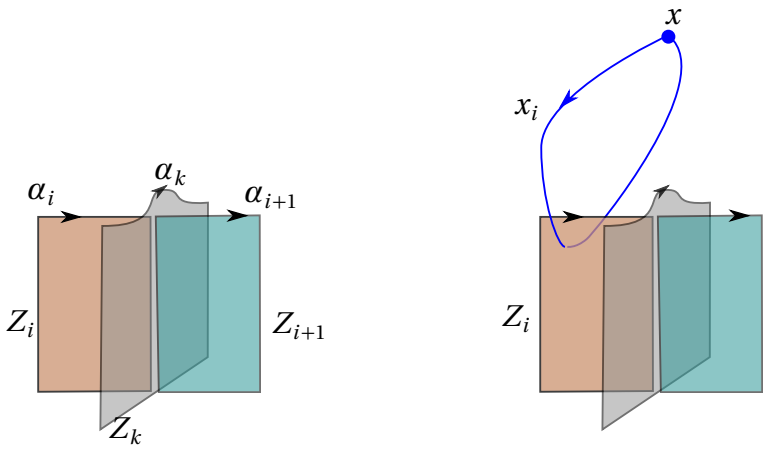


Figure 8.10: Half-planes under the strands

Any loop based at x not crossing any of the Z_i can be homotoped to the trivial loop, while any loop crossing the half-planes $Z_{i_1}, \dots, Z_{i_k}$ is homotopic to the concatenation of the loops $x_{i_1}, \dots, x_{i_k}$, maybe some of which are inverted depending on the orientations of the crossings. Thus the x_i generate the set of equivalence classes of loops. Furthermore, locally, a homotopy changing these generators either crosses a Z_i in two places, corresponding to a subword $x_i x_i^{-1}$, or it crosses a double intersection line of three half-planes Z_i, Z_{i+1} and Z_k, leading to modifying the word by one of the relations. □

So, fundamental groups of (complements of) knots are easy to compute, but this is hard to exploit in order to distinguish knots. First, it is not true that non-equivalent knots have non-isomorphic groups. Nevertheless, it is true for the unknot (it is the only knot corresponding to group $\mathbb{Z}$), and one can add additional structure to knot groups (using *peripheral systems*, see [BZ85, Section 3.C]) so that they distinguish all the knots. However, and more importantly, we also hit the same difficulty that we met when studying surfaces: while studying the group structure of $\pi_1(M)$ leads to a very rich algebraic structure and theory, it is hard to use it to extract algorithms, since most computational problems on groups presentations are undecidable in general. The following exercise shows that even the simplest cases require some work.

Exercise 8.3.3. Compute a presentation of the fundamental group of the complement of the figure-eight knot, and deduce from it that it is not trivial. *Hint:* Try mapping into a non-abelian finite group.

Some positive algorithmic results can be achieved by using the special structure of these groups (as was the case for surfaces). For example, the idea of finding a non-abelian representation which underlies the exercise above can be extended to work with any non-trivial knot, ultimately providing a polynomial-sized certificate of a knot being *not* the unknot (or equivalently that UNKNOT RECOGNITION is in **co-NP**), provided the Generalized Riemann Hypothesis is true [Kup14]. But this line of work falls largely outside of the scope of this class. We refer to the survey [AFW15a] for more information on this topic. Let us just mention one doomed idea in the next subsection.

8.3.2 Homology

If you did the exercises in the lecture notes on surfaces, you might recall Exercise 4.8 where you were asked to prove that fundamental groups of non-homeomorphic surfaces are not isomorphic. Viewed from another angle, this could be seen as a way to tell that two surfaces are not homeomorphic, similarly as what we are trying to do here, one dimension higher. One simple way to carry this out is to *abelianize* these groups (see the notes on minimum weight bases) and observe that the resulting abelian groups have different ranks. As we saw in the course on minimum weight bases, the abelianization of the fundamental group is the same thing as the first *homology group* $H_1(M)$. Since these groups are abelian, they are much more tractable algorithmically, and maybe they are strong enough to distinguish knot complements, and thus knots. This is not the case.

Proposition 8.3.4. *For any knot K, we have $H_1(\mathbb{S}^3 \setminus N) = \pi_1^{ab}(\mathbb{S}^3 \setminus N) = \mathbb{Z}$.*

PROOF. The equality of the first homology group and the abelianization of the fundamental group is a general topological result known as the Hurewicz Theorem, we refer to Hatcher [Hat02, Section 4.2] for the proof. Then, starting from the Wirtinger presentation, we simply observe that all the relations are of the form $x_k x_i x_k^{-1} x_{i+1}^{-1}$ or $x_i x_k x_{i+1}^{-1} x_k^{-1}$, which after abelianization yield $x_i x_{i+1}^{-1}$. Therefore, when abelianizing, all the generators merge into a single one and we obtain the group $\mathbb{Z}$. □

So the first homology group of the knot complement will not help us much in distinguishing knots.

8.3.3 Triangulations

Similarly as the way surfaces could be described by gluing together disks, one can cut 3-manifolds into balls and describe how these are glued together. To keep things simple (but they will still be complicated enough), we restrict our attention to the 3-dimensional analogue of triangulations using tetrahedra, which are also called triangulations instead of tetrahedrizations. A triangulation T is the topological space obtained from a disjoint set of t tetrahedra $T = (T_1, \dots, T_t)$ by (combinatorially) gluing some pairs of two-dimensional faces of these tetrahedra; a gluing between two faces is specified by a bijection from the vertex set of the first face to the vertex set of the second face. As a result of these gluing, edges and vertices of tetrahedra are also identified; it is also allowed to glue two zero-, one-, or two-dimensional faces of the same tetrahedron.

Since in a 3-manifold with boundary, the neighborhood of every point has to be an open ball or a half ball, the following conditions are necessary for a triangulation T to be a 3-manifold (possibly with boundary):

1. Each vertex has a neighborhood homeomorphic to $\mathbb{R}^3$ or to the closed half-space;
2. After the gluings, no edge is identified to itself in the reverse orientation.

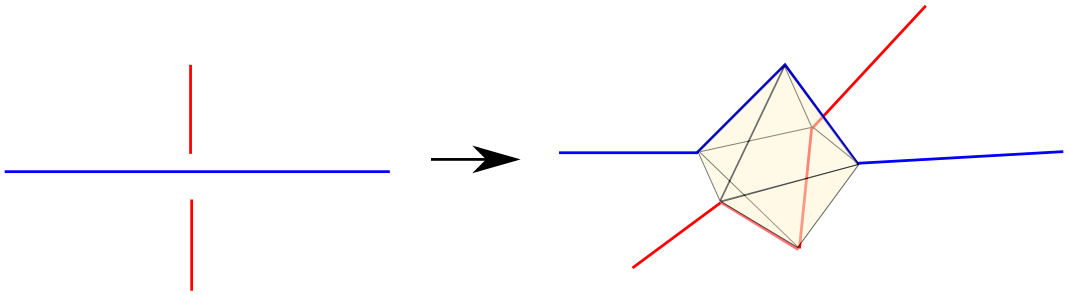


Figure 8.11: Putting an octahedron at every crossing.

Conversely, it is known [Moi52] that any 3-manifold M is the underlying space of such a triangulation (this is the 3-dimensional case of the Kerékjártó-Radó theorem that we saw for surfaces).

It is not very hard, yet somewhat tedious to build a triangulation of a knot complement starting from the knot.

Proposition 8.3.5. *Given a polygonal knot K made of n segments, we can compute a triangulation of $\mathbb{S}^3 \setminus N$ in $O(n)$ time.*

PROOF. Starting with a polygonal knot, project it into a knot diagram, and triangulate the resulting planar graph. Then at each crossing, use the octahedron gadget of Figure 8.11. Now, outside the octahedra, there is a big 3-dimensional polyhedron, and after subdividing it into tetrahedra, we obtain a triangulation of $\mathbb{S}^3$ where K lies on the edges of the tetrahedra. Drilling tubes around the corresponding edges and retriangulating the space gives a triangulation of $\mathbb{S}^3 \setminus N$. □

For surfaces, it was somewhat easy to recognize which surface one would obtain by identifying the disks by just visualizing the corresponding identifications in 3 dimensions. Since our imagination is much more lacking with 4-dimensional space, the corresponding approach to recognize 3-manifolds via their triangulations is much harder, as the following mind-bending exercise dealing with a single tetrahedron (!) with a single vertex (!!) that can actually be embedded in $\mathbb{R}^3$ (!!!) showcases.

Exercise 8.3.6. Take a single tetrahedron and label its vertices by $0, 1, 2$ and 3. Identify the 012 face with the 130 face by sending the vertices $0, 1$ and 2 respectively to $1, 3$ and 0.

1. Prove that the resulting space is a 3-manifold with boundary.
2. Which familiar one is it?

8.4 An algorithm for unknot recognition

Now that we have seen many approaches that do not provide algorithms, our goal in this section is to present an algorithm for UNKNOT RECOGNITION, due to Hass, Lagarias and Pippenger [HLP99] (following ideas of Haken [Hak61]) which shows that UNKNOT

RECOGNITION is in **NP**. More broadly, this algorithm is an illustration of the power of normal surface theory, which is an ubiquitous tool in the study of computational problems in low dimensions.

8.4.1 Normal surface theory

A **normal surface** in T is a properly embedded surface in T that meets each tetrahedron in a possibly empty collection of triangles (cutting off a vertex) and quadrilaterals (separating a pair of vertices), which are called *normal disks*. In each tetrahedron, there are 4 possible types of triangles and 3 possible types of quadrilaterals, pictured in Figure 8.12. The intersection of a normal surface with a face of the triangulation gives rise to a **normal arc**. There are 3 possible types of normal arcs within each face: the type of a normal arc is defined according to which vertex of the face it separates from the other two.

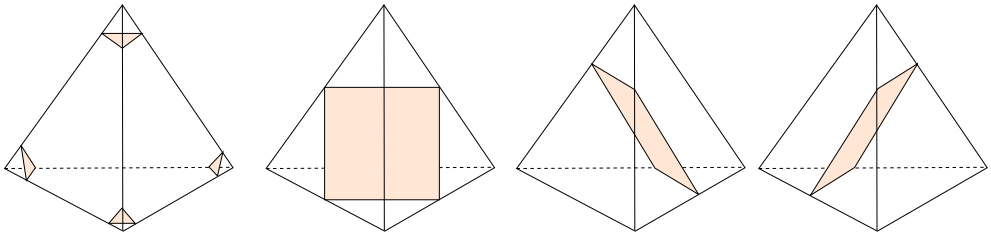


Figure 8.12: The seven types of normal disks within a given tetrahedron: Four triangles and three quadrilaterals.

With each normal surface S, one can associate a vector, denoted by $[S]$, in $(\mathbb{Z}_+)^{7t}$, where t is the number of tetrahedra in T, by listing the number of triangles and quadrilaterals of each type in each tetrahedron. This vector provides a very compact and elegant description of that surface.

The vector $[S]$ corresponding to a normal surface S, called its **normal coordinates**, satisfies two types of conditions:

- The first type of conditions is the **matching equations**. Consider a normal arc type in a given non-boundary face f of T. This normal arc type corresponds to exactly one triangle normal coordinate, $v_{t,1}$, and one quadrilateral normal coordinate, $v_{q,1}$, in a tetrahedron incident with f. Similarly, let $v_{t,2}$ and $v_{q,2}$ be the triangle and quadrilateral normal coordinates corresponding to that arc type in the opposite tetrahedron. The matching equation for that arc type is $v_{t,1} + v_{q,1} = v_{t,2} + v_{q,2}$. Intuitively, this means that for at a face between two tetrahedra, there are as many objects going in as objects going out. There are no matching equations for faces on the boundary of the triangulation.

- The second type of conditions, the **quadrilateral conditions**, stipulates that, within any tetrahedron, at most one of the three quadrilateral coordinates must be non-zero. Indeed, two quadrilaterals of different types within the same tetrahedron must cross, and therefore this condition is needed to ensure that the surface does not self-intersect.

Conversely, if T is a triangulation of size t and v is a vector in $(\mathbb{Z}_+)^{7t}$, then v corresponds to a normal surface if and only if the matching equations and the quadrilateral conditions are fulfilled. The reconstruction process can be depicted as follows:

- In each tetrahedron, by the quadrilateral conditions, there is at most one type of quadrilateral. One places as many parallel copies of this quadrilateral as needed in the tetrahedron, and then place the parallel triangles next to every vertex of the tetrahedron. It is straightforward to do so without having any intersection between triangles and quadrilaterals.
- One glues the faces on the triangulation together, and in the process, one needs to glue normal arcs, i.e., triangles or quadrilaterals on the one side to triangles and quadrilaterals on the other side. By the matching equations, the numbers fit, and the gluing is imposed by the order in which the normal disks are placed in the tetrahedra.

A **normal isotopy** is an ambient isotopy of M that fixes globally each vertex, edge and face of T. Therefore, a normal surface is represented up to a normal isotopy by a vector in $(\mathbb{Z}_+)^{7t}$ satisfying the matching equations and the quadrilateral conditions. Moreover, given a triangulation and normal coordinates, checking that the matching equations or the quadrilateral conditions hold can trivially be done in linear time.

From this construction, one sees moreover that every vector of normal coordinates corresponds to a unique normal surface, up to a normal isotopy. Therefore, we will often abuse the notation and call both S and $[S]$ a normal surface.

Finally, if one is given normal coordinates, how can one recognize which surface it corresponds to? For usual surfaces it is enough to compute the Euler characteristic and check for orientability, but there is an issue of compression here: a vector of complexity n can correspond to a normal surface with 2^n normal disks, and thus computing the Euler characteristic "by hand" would take an exponential time. But one can do better:

Lemma 8.4.1. *There exists a linear form e on $\mathbb{Z}_+^{7t}$ such that if $[S]$ is the coordinate vector of a normal surface S, $e([S]) = \chi(S)$ where χ is the Euler characteristic.*

PROOF. A normal coordinate describes the number of normal triangles or quadrilaterals of some type in a tetrahedron. One can estimate how much such a triangle or quadrilateral contributes to the Euler characteristic of the entire surface. Let us pick a triangular normal disk t adjacent to edges e_1, e_2 and e_3 of T, and let us denote by v_i the valence of edge e_i, i.e., the number of tetrahedra around it. Then the contribution of t to the Euler characteristic of S is $1/v_1 + 1/v_2 + 1/v_3 - 3/2 - e_\partial/2 + 1$, where e_∂ is the number of edges of t on the boundary of the manifold. Indeed, the vertices of t (on the edges e_1, e_2 and e_3) contribute 1 but are "shared" between all the normal disks adjacent to them. Similarly, the edges contribute -1 but are shared between two normal disks (except for the boundary ones), and the face contribution is exactly once. For a quadrilateral, we get $1/v_1+1/v_2+1/v_3+1/v_4-4/2-e_\partial/2+1$. Summing all of these contributions provides a linear form e which matches with the Euler characteristic. □

8.4.2 Trivial knot and spanning disks

The apparatus of normal surface theory is designed to study the surfaces embedded in a 3-manifold. In the case of knot complements, this is very relevant to unknot recognition because of the following easy lemma.

Lemma 8.4.2. *A knot K is trivial if and only if it is the boundary of an embedded disk.*

PROOF. One direction is immediate: if K is trivial, then it is ambient isotopic to the usual embedding of $\mathbb{S}^1$, which obviously bounds a disk. An ambient isotopy is a homeomorphism, thus it preserves this disk, so the knot K also bounds a disk.

The other direction is not much harder: if a knot K bounds a disk, then there is an ambient isotopy that contracts K progressively along this disk until it lies in an arbitrarily small neighborhood of a point, where it will be ambient isotopic to a usual embedding of $\mathbb{S}^1$. □

The starting idea of the algorithm for UNKNOT RECOGNITION is to simply find this disk if it exists. But the triangulation we will work with $\mathbb{S}^3 \setminus N(K)$ instead of just $\mathbb{S}^3$ with K in its 1-skeleton. This is very much needed in order to apply normal surface theory: indeed, we will want to find this disk as a normal surface, and normal surfaces are by construction *transverse* to the 1-skeleton of the triangulation. Therefore, one needs to drill a small tube around K in order to use normal surfaces, we denote this small tube by T_K. The corresponding lemma that we will need is the following one.

Lemma 8.4.3 ([HLP99, Lemma 4.1])**.** *A knot K is trivial if and only if there exists a disk D in $\mathbb{S}^3 \setminus N(K)$ such that the boundary of D is non-trivial in ∂T_K.*

PROOF. The proof works similarly: If K is trivial, there is an ambient isotopy carrying it to the standard embedding of $\mathbb{S}^1$ into $\mathbb{R}^3$. This ambient isotopy preserves the trivializing disk and the homotopy class of its boundary on ∂T_k, which is therefore non-trivial.

The other direction requires some more work. A **meridian** of a knot is a simple closed curve on T_K bounding a disk in T_K. A **longitude** of a knot is a simple closed curve on T_K crossing the meridian exactly once, and inducing a null-homologous curve in $\mathbb{S}^3 \setminus N(K)$. The existence of these curves follows, for example from the computation of $H_1(\mathbb{S}^3 \setminus N(K))$ in Section 8.3.2.

If K is non-trivial, let us assume by contradiction that there exists a disk D having boundary γ non-trivial on ∂T_K. The homology of γ on ∂T_K is $a_1[m] + a_2[\ell]$ with $a_1, a_2 \neq (0,0)$ where $[m]$ and $[\ell]$ denote the homology classes induced on ∂T_K by the meridian and the longitude, respectively. Via the inclusion map $\partial T_k \hookrightarrow \mathbb{S}^3 \setminus N(K)$, γ has a homology class in $H_1(\mathbb{S}^3 \setminus N(K))$, which is a_1 by definition of the meridian and the longitude. Since γ bounds a disk, we thus have $a_1 = 0$, and thus $a_2 = \pm 1$ by simplicity of γ. Therefore K cobounds an annulus with γ within T_K. Gluing this annulus with the disk D, we obtain a disk bounded by K and we can apply the previous lemma. □

Let us call a disk satisfying the above properties a **spanning disk**. The key theorem behind the **NP** algorithm is the following one.

Theorem 8.4.4. *Let K be a trivial knot and T be a triangulation of $\mathbb{S}^3 \setminus N(K)$ obtained by the process of Section 8.3.3. Then there exists a spanning disk of K that is normal with respect to T and of which normal coordinates are bounded by $2^{O(t)}$.*

Assuming Theorem 8.4.4 for now, we can (almost) provide the advertised algorithm. Naively, this should be very simple: now that we have established that there exists a spanning disk that is normal and has coordinates of bounded size, one can simply use this spanning disk (or rather its normal coordinates) as a certificate. The size of the coordinates might be exponential, but exponential numbers can be encoded using only a polynomial number of bits (this trivial observation is what makes everything work!), thus the certificate has polynomial size. But there is a hidden issue lurking here: one also needs to verify in polynomial time that the certificate is indeed a spanning disk. This should be easy in principle, but as we have already mentioned, the certificate is very **compressed** due to this integer encoding, and it turns out to be non-trivial.

Corollary 8.4.5. UNKNOT RECOGNITION *is in* ***NP****.*

PROOF. Starting with a knot K, described by a diagram with n crossings, one first builds a triangulation of its complement following Section 8.3.3. This triangulation has $O(n)$ tetrahedra. The certificate is then the normal spanning disk promised by Theorem 8.4.4. It is a vector in $O(n)$ dimensions and the size of the coordinates is bounded by $2^{O(n)}$, which can be encoded with a number of bits polynomial in n.

Once one is given the certificate, one can verify that is it indeed a spanning disk in the following way:

1. One first verifies that this is indeed a normal surface S, by checking that the matching and quadrilateral constraints are satisfied.
2. One checks whether S is connected.
3. One checks whether S is a disk.
4. One checks whether the boundary of S is non-contractible.

If all the answers are positive, then we have a spanning disk and K is unknotted. Step 1 can be easily done in polynomial time since the matching equations are linear, and the quadrilateral constraints are easy to verify by looking at each coordinates. Let us assume that Step 2 has been done for now. In order to do step 3, it is enough to compute the Euler characteristic of S, which can be done in polynomial time using the linear form of Lemma 8.4.1. For step 4, from the normal coordinates of S one can obtain normal coordinates[3] of ∂S. Since the fundamental group of the torus is $\mathbb{Z}^2$, testing whether ∂S is contractible can be done in polynomial time from these normal coordinates.

[3]More formally, ∂S is a curve on a torus, and the coordinates we obtain describe this curve with respect to the triangulation of the torus inherited from T. The underlying theory of **normal curves** is similar (but much simpler) than for normal surfaces.

However, Step 2 is hard. The naive algorithm, which would just follow a starting normal disk and count the resulting connected components could require *exponential* time, since the size of the normal coordinates could be exponential. A polynomial algorithm to check connectivity of normal surfaces was designed by Agol, Hass and Thurston [AHT06] and can be used here, but we will not delve into this. □

The rest of this section is devoted to the proof of Theorem 8.4.4. We first show how to normalize a spanning disk, and then how to prove the existence of one with suitably bounded coordinates.

8.4.3 Normalization of spanning disks

A key observation, initially due to Haken [Hak61], is that there exists a spanning disk that is a normal surface:

Lemma 8.4.6. *Let K be a trivial knot and T be a triangulation of $\mathbb{S}^3 \setminus N(K)$ obtained by the process of Section 8.3.3. Then there exists a spaning disk that is normal with respect to T.*

PROOF. The proof proceeds by starting with a spanning disk D (which exists by Lemma 8.4.3) and *normalizing* it. We can first, by using a small perturbation, put D in general position with respect to the triangulation T: i.e., all the intersections between D and T can be assumed to be transverse. Now, we look at what can go wrong, i.e., which pieces in D are not normal with respect to T. Let us write $c(D) = (\mathrm{wt}_1(D), \mathrm{wt}_2(D))$, where wt_1 and wt_2 respectively denote the number of connected components of the intersection of D with the edges of T and the faces of T, and order the pairs $c(D)$ with the lexicographic order. There are many different occurences of non-normality which we deal with distinct moves, but each time the complexity $c(D)$ will only go down. Since this complexity is finite to begin with, after a finite number of steps, the process will finish and we will have a normal surface.

1. If D intersects an inner face F of T in an arc that hits twice the same interior edge e of T, one can push D locally to reduce its number of intersections with e, see Figure 8.13.

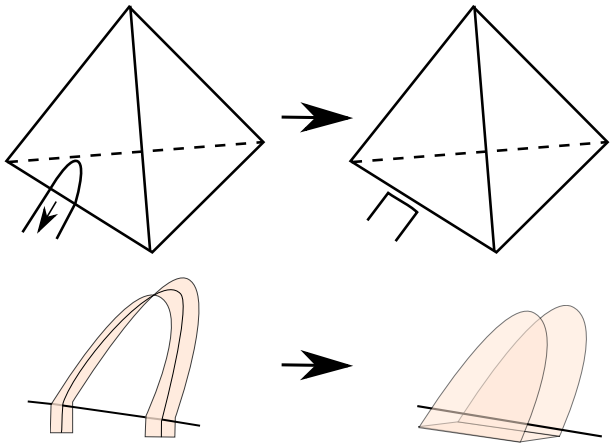


Figure 8.13: If there are excess intersections with an interior face and an interior edge, one can push the disk to reduce its intersections with the edges of T. The first picture shows what the move induces on the face F and the second one is the corresponding 3-dimensional move.

2. If D intersects a boundary face F of T in an arc that hits twice the same boundary edge e of T, one can push D similarly, reducing its number of intersections with e, see Figure 8.14. The boundary of D is moved by this operation, but only by a homotopy, hence the resulting disk is also a spanning disk.

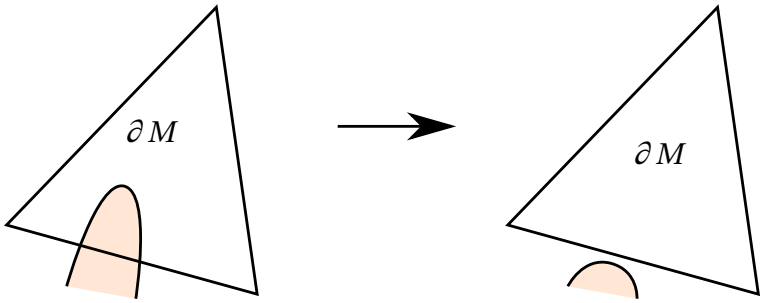


Figure 8.14: If there are excess intersections with a boundary face, one can push the disk to reduce its intersections with the edges of T

3. If D intersects a face F in a cycle disjoint from its edges (forming locally a tube), one can "cut" (or **compress**) this tube, which will reduce the number of intersections of D with F. Such a compression cuts the disk into a sphere and a disk, and one can continue the normalizing process with the disk, which will have smaller complexity, see Figure 8.15.

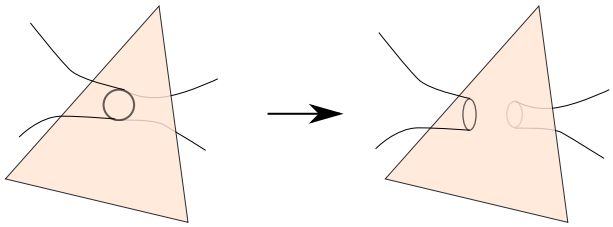


Figure 8.15: If the disk forms a tube crossing a face of T, one can cut this tube and keep one of the components to reduce the number of intersections with the faces of T.

4. If D intersects a tetrahedron t in a tube (i.e., locally, D intersects ∂t in two components), then one can compress this tube as well. For the same reasons, by picking the remaining disk, the complexity will have decreased, see Figure 8.16.

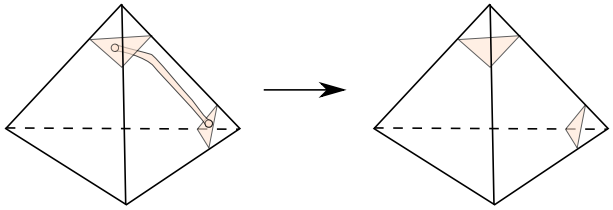


Figure 8.16: If the disk forms a tube inside a tetrahedron, one can cut this tube and keep one of the components to reduce the number of intersections with the edges of T.

5. If D intersects an inner face F of T in an arc a that hits twice the same boundary edge e of T, one can "cut" (or **boundary compress**) along the disk that is bounded by a and e. This has the effect of cutting the disk D into two subdisks, one of which must be spanning. Taking this one reduces the complexity, see Figure 8.17.

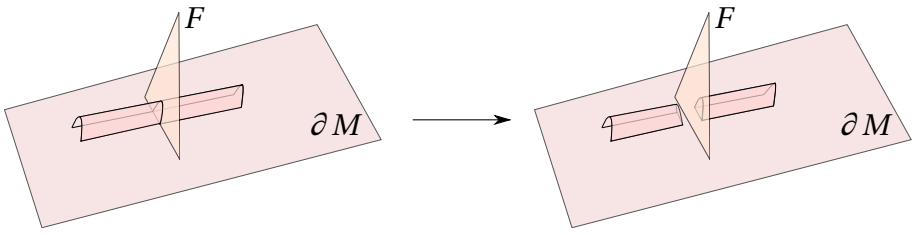


Figure 8.17: If there are excess intersections with an interior face and a boundary edge, one can push the cut the disk and keep one of the components to reduce its intersections with the edges of T.

6. Finally, D might be locally too complicated inside a tetrahedron t, i.e., $D \cap t$ can have more than 4 arcs. In this case one can show that there are at least 8 arcs, and some edge is hit at least twice. Then, if that edge is not on the boundary of the triangulation, one can reduce the complexity by pushing D towards that

edge, see Figure 8.18. Otherwise, one can do a boundary compression as in the previous step.

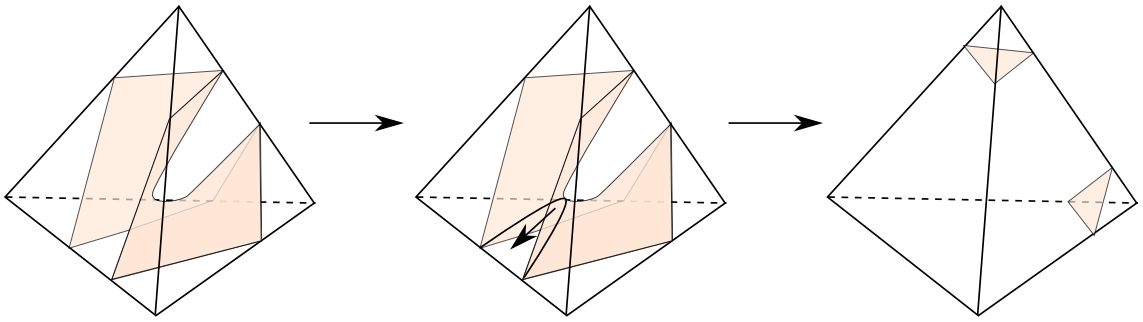


Figure 8.18: If there is a piece inside a tetrahedron with more than 4 arcs, one can push it to simplify it.

Note: For all these moves, there might be other pieces of the disk in the way of the indicated move. This is resolved by always applying first the moves corresponding to an innermost disk, i.e., first normalizing the non-normal behavior closest the the boundary of the tetrahedron.

Once none of these non-normal cases happen, the disk D intersects the triangulation T in a normal way, hence we have found a normal spanning disk. □

We have thus established that it suffices to find a normal spanning disk to certify that a knot is trivial. Since normal surfaces can be described by a vector in $\mathbb{Z}_+^{7t}$, this would give an algorithm if one could bound the size of these coordinates. Such a bound will be established by exploiting the additive structure on normal surfaces provided by these vectors.

8.4.4 Haken sum, fundamental and vertex normal surfaces

The set of vectors of $\mathbb{R}_+^{7t}$ verifying the matching equations is called the **Haken cone** $\mathscr{C}$. The normal surfaces are the integral points in this cone that also satisfy the quadrilateral constraints. It they have no conflicting quadrilaterals, two normal surfaces can be added by adding their vectors, and the result will still be a normal surface since the matching equations are linear. This operation is called the **Haken sum** of normal surfaces. Note that by Lemma 8.4.1, if S is the Haken sum of two normal surfaces S_1 and S_2, $\chi(S) = \chi(S_1) + \chi(S_2)$.

A normal surface $[S]$ is called **fundamental** if it can not be written as a sum $[S] = [S_1] + [S_2]$ with $[S_1]$ and $[S_2]$ two non-empty normal surfaces. A fundamental normal surface $[S]$ is a **vertex normal surface** if $c[S] = c_1[S_1] + c_2[S_2]$ for positive integers c, c_1 and c_2 implies that $[S_1]$ and $[S_2]$ are multiples of $[S]$. Fundamental and vertex normal surfaces are the building blocks for normal surfaces, and crucially, one can bound their complexity:

Lemma 8.4.7. • *Let $[S]$ be a vertex normal surface in a triangulation T with t tetrahedra. Then the normal coordinates of S have size bounded by $2^{O(t)}$.*

- *Let $[S]$ be a fundamental normal surface in a triangulation T with t tetrahedra. Then the normal coordinates of S have size bounded by $2^{O(t)}$.*

PROOF.
- Let us intersect the cone $\mathscr{C}$ with the hyperplane $H = \sum_i x_i = 1$. This forms a polyhedron $\mathscr{P}$ and the vertex normal surfaces will be obtained as the first integral multiples of some of the vertices of $\mathscr{P}$. Now, the vertices of $\mathscr{P}$ are obtained as a solution of $7t$ equations, which either come from the matching equations, from the hyperplane H or from a hyperplane of the form $x_i = 0$. Thus, such a vertex v verifies $Mv = (0,\dots,0,1)^T$ for some matrix M with entries in $\{-1,0,1\}$. By Cramer's rule, the coordinates v_i are obtained by the quotient $\det M_i / \det M$, where M_i is the matrix M where the ith column has been replaced by $(0,\dots 0,1)^T$. One can bound these determinants using Hadamard's inequality $(\det M)^2 \leq \prod_i \|r_i\|^2$ where r_i are the rows of M. We obtain $|\det M_i| = 2^{O(t)}$ and $|\det M| = 2^{O(t)}$, and thus $v_i = 2^{O(t)}$. Then, the size of the coordinates of the vertex normal surfaces is bounded by $v_i |\det M| = 2^{O(t)}$ as well.
- Let $[S]$ be a fundamental normal surface, then multiples of $[S]$ can be decomposed on the vertex normal surfaces: $c[S] = \sum c_i[S_i]$, or equivalently $[S] = \sum c_i/c\,[S_i]$. Note that $c_i/c \leq 1$, otherwise one would have $[S] = ([S]-[S_i])+[S_i]$ which would be a non-trivial integral decomposition of S, a contradiction. Thus any coordinate of $[S]$ is at most the sum of the coordinates of the vertex normal surfaces $[S_i]$, of which there are at most $2^{O(t)}$ (one can for example bound the number of matrices M involved in the previous item). This concludes the proof.

□

A common principle in normal surface theory is that "interesting" surfaces in a 3-manifold can be found among the fundamental normal surfaces, and even sometimes among the vertex normal surfaces of the triangulation. This turns out to be true for spanning disks.

Proposition 8.4.8. *Let K be a trivial knot and T be a triangulation of $\mathbb{S}^3 \setminus N(K)$ obtained by the process of Section 8.3.3. Then there exists a spanning disk that is a fundamental normal surface with respect to T.*

Combining Lemma 8.4.7 and Proposition 8.4.8 directly proves Theorem 8.4.4, and thus the **NP** algorithm (modulo the connectivity issue already mentioned).

Remark: The main issue in the **NP** algorithm is to check connectivity in polynomial time. One way to circumvent it could be to verify that the certificate describes a fundamental normal surface, since fundamental normal surfaces are connected. But there is no easy way to do that either. On the other hand, one can certify that a normal surface is a vertex normal surface, by exhibiting the family of $7t$ equations it satisfies (see the proof of Lemma 8.4.7). And it is also true, but harder to prove, that there exists a spanning disk that is a vertex normal surface: we refer to Jaco and Tollefson [JT95] for a proof. Thus, if one admits this, it gives an alternative way to provide a **NP** certificate.

There remains to prove Proposition 8.4.8.

PROOF OF PROPOSITION 8.4.8. The main idea of the proof is to use the Euler characteristic as an accounting device. Indeed, since the Euler characteristic is linear on the normal coordinates, one can use it to discard the vast majority of the bad cases, and the last ones will be handled by hand. More precisely, let D be a normal spanning disk. We can assume that its boundary crosses at most once every triangle of ∂M and that it is of minimal complexity subject to this. If it is not fundamental, it can be written as a sum $[D] = [S_1] + [S_2]$ where the $[S_i]$ are non-trivial normal surfaces, and thus $\chi(S_1) + \chi(S_2) = 1$. Among all the decompositions into S_1 and S_2, let us pick the one that minimizes the number of connected components of $S_1 \cap S_2$.

We first claim that that S_1 and S_2 are connected. Indeed, if S_1 is not connected and consists of two disjoint connected components A and B, since D is connected, S_2 intersects both A and B. But then $[D] = [S_2 + A] + [B]$ and this decomposition has less intersections than S_1 and S_2, contradicting our assumption.

Then, since the Euler characteristic of a connected surface is at most 2, there are only a few cases to deal with. The favorable one is when S_1 is a disk and S_2 is a torus, then S_1 has the same boundary as D, and thus is a spanning disk of smaller complexity, a contradiction. There remains to discard the bad cases (note that since we are in $\mathbb{R}^3$, unpunctured projective planes and Klein bottles can be ruled out straight away) :

1. S_1 is a punctured torus or Klein bottle and S_2 is a sphere.
2. S_1 is a Möbius band or an annulus and S_2 is a disk.

In order to do so, we must pause a bit and figure out what a Haken sum means geometrically: if S_1 and S_2 are two intersecting normal surfaces, their Haken sum is obtained by taking the normal disks of S_1 and S_2 and "reconnecting" them differently. This amounts geometrically to looking at the intersection curves between S_1 and S_2, cutting along them and doing a **switch**, as pictured in Figure 8.19. There are two possibilities for a switch, and the one carried out depends on the situation of the intersecting normal disks inside a tetrahedron : both switches gives surfaces, but only one gives a normal surface. But if one performs the "bad" switch, one can still normalize the resulting surface to obtain a normal surface.

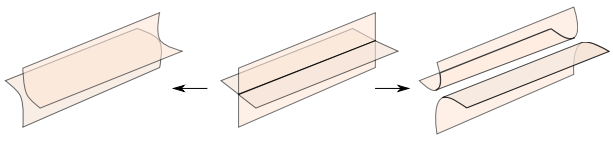


Figure 8.19: The two switches at an intersection curve.

For the first case, let α be a curve of intersection between S_1 and S_2. We first claim that α is not separating in the punctured torus S_1. Indeed, otherwise, in the case where S_1 is a punctured torus, by cutting S_1 along α and performing the switch that patches a disk of S_2 bounded by α on this cut, and normalizing if necessary, one would obtain either a spanning disk of less complexity, or a decomposition of D into two normal surfaces which intersect less than S_1 and S_2, contradicting our assumptions. If S_1 is

a punctured Klein bottle, there is a third case if α cuts the Klein bottle into a Möbius band and a Möbius band with an additional boundary, but pasting the Möbius band on the disk of S_2 yields an immersion of the projective plane in $\mathbb{R}^3$ without triple points, which is impossible [Ban74].

Now, since S_1 intersects S_2 along at least two non-contractible curves, and we pick the two outermost such curves, i.e., the pair of non-contractible curves closest to ∂S_1 on S_1, see Figure 8.20. When one cuts along these curves and glue disks coming from S_2, one obtains a disk with boundary ∂S_1. Since this cut-and-pasting corresponds to a (good or bad) switch, after normalizing if needed, we obtain a spanning disk of lower complexity than D, a contradiction.

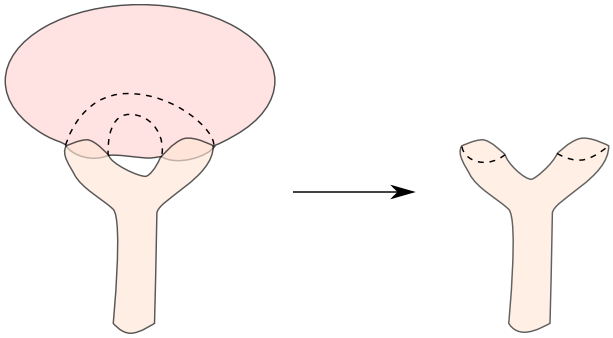


Figure 8.20: One can cut the torus S_1 along non-contractible cycles and patch it with disks of S_2 to obtain a spanning disk of lower complexity.

In the second case, ∂S_2 and ∂S_1 cross each other, which cannot happen since ∂D crosses at most once each triangle of ∂M. Thus case 2 is ruled out, and this concludes the proof. □

8.5 Knotless graphs

To conclude this chapter with a striking open problem, let us discuss a bit about knotless graphs. A polygonal embedding of a graph G into $\mathbb{R}^3$ is **knotless** if every simple cycle of G is mapped to a trivial knot by this embedding. A graph G is **knotless** if it admits a knotless embedding, and **intrinsically knotted** otherwise. It is not obvious that there exist intrinsically knotted graphs at all (remember that is was not obvious that there existed non-trivial knots either), but one can prove, for example using the *Arf invariant* that K_7 is intrinsically knotted [CG83].

As we discussed in this chapter, no polynomial-time algorithm for UNKNOT RECOGNITION is known. Therefore, recognizing whether a given embedding of graph is knotless in polynomial time is also out of reach for current techniques. One could expect recognizing knotless graphs to be even harder since naively, it amounts to making this test for every possible embedding of G into $\mathbb{R}^3$. Therefore the following proposition might come as a shock.

Proposition 8.5.1. *There exists an algorithm to recognize knotless graphs in polynomial time.*

PROOF. If H is a minor of G and G is knotless, H is knotless as well: if i is a knotless embedding of G, every simple cycle of H corresponds to a simple cycle of G and is thus mapped by i to a trivial knot. Thus knotless graphs form a minor-closed family, and it follows from Robertson-Seymour theory (see for example [Lov05] for an introduction) that they can be recognized in polynomial time. □

The proof of this proposition is shockingly unsatisfying: not only is the algorithm, as most algorithms coming from Robertson-Seymour theory, extremely inefficient, but we actually *do not know* what it is: what the theory proves is that minor-closed families are characterized by a finite family of forbidden minors, and testing for forbidden minors can be done in polynomial time – but we do not know what these are. It is an open problem to find an *explicit* polynomial-time algorithm to recognize knotless graphs.

9

Undecidability in Topology

Contents

The purpose of this lecture is to make explicit the limits of computational topology by showing that some simple and natural questions in topology are undecidable. In order to make the statement precise we need to define the notion of decidability and to specify the description of topological spaces we are interested in. Concerning topological spaces we should consider spaces having a combinatorial description such as finite simplicial complexes[1]. Note that many interesting spaces have such a description: compact topological manifolds of dimensions 2 or 3, compact differentiable manifolds, etc. See [Man14] for a survey. Concerning decidability there are essentially two notions. One refers to the independence of a statement with respect to a logical system. In other words, the statement is undecidable if neither its affirmation nor

[1] Recall that a simplicial complex is a collection of simplices glued in a nice fashion. It is the total space of an *abstract simplicial complex* over a set V, which is a family of subsets of V closed under the operation of taking non-empty subsets.

its negation can be proved from the axioms of the system using its logical rules. The existence of such undecidable statements relates to the first Gödel's incompleteness theorem. The other notion of decidability refers to a family of problems with YES/NO answers, such as testing a property over a family of objects, and expresses the existence of an algorithm to output the answer of any problem in the family. Note that any finite family of problems for which the answers is provable is always decidable in this acception. Indeed, an algorithm to solve the problems just needs to store the correct answer of each problem. Paradoxically, this is valid even if we do not know yet the correct answers since the decidability only claims the existence of an algorithm and not the algorithm itself.

Both notions of decidability may be relevant to computational topology. As an illustration, consider the contractibility problem of deciding if a closed path can be continuously deformed into a point in a simplicial complex. We will prove that there is no algorithm to decide this problem given the path and the simplicial complex as input. As a stronger statement there exists a simplicial complex for which there is no algorithm that decides the contractibility of the closed paths in this simplicial complex. At last, there exists a closed path in some simplicial complex for which it cannot be logically decided if the path is contractible or not.

Most often, undecidability results in topology are shown by first transforming a decision problem into a question concerning combinatorial group theory. In turn, problems about groups are transformed into problems about Turing machines. Ultimately, the proofs of undecidability rely on a reduction to the halting problem for Turing machines. We recommend the survey by Poonen [Poo14] for many undecidable problems in mathematics.

9.1 The Halting Problem

9.1.1 Turing Machines

A **Turing machine** is a mathematical model for the notion of computation. It was introduced by Alan Turing in 1936. According to Church-Turing thesis this is a universal model for the mechanization of computation. It was proved equivalent to other notion of computation such as recursive functions and λ-calculus.

Formally, a Turing machine is a triple $(\mathscr{A}, \mathscr{Q}, \mathscr{T})$, where $\mathscr{A}$ is a finite alphabet including a special **blank** character, $\mathscr{Q}$ is a finite set of **states**, and $\mathscr{T} \subset \mathscr{A} \times \mathscr{Q} \times \mathscr{A} \times \mathscr{Q} \times \{R, L\}$ is a **transition table** specifying how the machine operates on **configurations**. Those are words of the form $uqv \in \mathscr{A}^* \times \mathscr{Q} \times \mathscr{A}^*$. Such a configuration represents the machine in state q together with a linear tape marked with the word uv and whose read/write head is on the first letter in v (the empty word is interpreted as a blank). Transition $aqbpD \in \mathscr{T}$ applies to any configuration uqv such that a is the first letter in v. It transforms uqv replacing a with b, the state q by p, and moves the head one step to the left or right according to whether D equals L or R, respectively.

From the computability perspective there is no loss of generality to consider **deterministic** machines for which $aqbpD \in \mathscr{T}$ and $aqb'p'D' \in \mathscr{T}$ implies $b' = b$, $p' = p$ and $D' = D$: reading a letter in some state leads to only one new possible configuration. The machine is **halting** in a given configuration when no transition applies.

Standard coding of Turing machines

A Turing Machine M is in **standard form** if its alphabet is a finite subset of $\Sigma = \{\text{blank}, 1, 1', 1'', 1''', \ldots\}$ and its set of states is a finite subset of $\{q, q', q'', q''', \ldots\}$. One can encode the transition table of M on the six letter alphabet $\{\text{blank}, 1, q, ', R, L\}$ by concatenating its transitions (of the form $1'q'1''q''D$), where the prime symbol is considered as a letter. Finally, replacing $q, ', R, L$ by the respective letters $1', 1'', 1''', 1''''$, we obtain a coding of the transition table over the finite alphabet $\{\text{blank}, 1, 1', 1'', 1''', 1''''\}$. This coding is the **standard code** of M and is denoted by $\lceil M \rceil$.

9.1.2 Undecidability of the Halting Problem

A set of words $W \subset \mathscr{A}^*$ is **decidable**, or **recursive**, if there exists a turing machine $M = (\mathscr{A}, \mathscr{Q}, \mathscr{T})$ with three states $q_i, q_a, q_r \in \mathscr{Q}$, respectively called *initial, accepting* and *rejecting*, such that for every $w \in \mathscr{A}^*$ the machine M starting from configuration $q_i w$ reaches a halting configuration in state q_a if $w \in W$ and in state q_r otherwise. In particular M always reaches a halting configuration. Note that W is decidable if and only if both W and its complement $\mathscr{A}^* \setminus W$ are semi-decidable. Recall that W is **semi-decidable** if there exists a Turing machine halting in an accepting state if and only if it is given as input a word of W. Although this definition does not require any behavior for words not in W, it is equivalent to assume that the machine never stops given such words. Unfortunately, the same definition was given many names such as **semi-recursive**, **recursively enumerable**, **computably enumerable**, **listable** or **Turing recognizable**. The plurality of names comes from the fact that it is equivalent to require the existence of a Turing machine that enumerates W, *i.e.*, outputs all its words one after the other. A **decision problem** is a set of questions with YES/NO answers. By extension this problem is decidable, or **algorithmically solvable**, if the questions can be encoded as words over a finite alphabet and if the subset of words corresponding to questions with positive answers is decidable.

Consider the **self-halting problem** of deciding if a Turing machine M given as input its own standard code, *i.e.* starting with the configuration $q_i\lceil M \rceil$, will eventually reach a halting configuration in the accepting state.

Theorem 9.1.1. *The self-halting problem is semi-decidable but not decidable.*

PROOF. That the self-halting problem is semi-decidable is quite clear. Given the standard code of a Turing machine, it is enough to simulate the corresponding machine on this same input. The notion of universal Turing machine (see below) provides such a simulation. By way of contradiction, suppose that the self-halting problem is decidable. Hence, there exists a Turing machine, say S, that recognizes the complementary language. In other words, S halts in the accepting case if the input *does not* correspond to the standard code of a Turing machine that halts in the accepting state on its own input, and runs forever otherwise. Let us run S with the initial configuration $q_i\lceil S \rceil$. If S halts in the accepting state, this means that S does not halt in the accepting state on its own input, a contradiction. So S must run forever, meaning that S does halt in the accepting state on its own input, and we have again reached a contradiction. □

The general **halting problem** is to decide, given a machine M and a starting configuration I if M reaches a halting configuration. Since the self-halting problem is a particular case of the halting problem, we obtain:

Corollary 9.1.2. *The halting problem is unsolvable.*

Universal Turing Machine

A Turing machine T is said **universal** if for any Turing machine M and any initial configuration C, starting from configuration $q_i[M]C$ the machine T simulates the computation of M from C and halts in its accepting state if and only if this computation eventually stops. Though fastidious, one can write a program in his favourite language, say in C++, to simulate a universal Turing machine. This proves a fortiori its existence. The idea is to traverse the initial configuration C to "read" its state and the current symbol (the one that should lie under the reading head of M). Then, T needs to traverse $[M]$ in order to find the transition that applies. This transition transforms C into a configuration C' and we obtain the configuration $q_i[M]C'$ on T. We can proceed this way until some configuration $q_i[M]C''$ is reached, where C'' is a halting configuration for M. In this case, T should stop in its accepting state. Otherwise, T runs forever.

Theorem 9.1.3. *The halting problem for the universal machine T is unsolvable.*

In other words, there is no Turing machine that can decide for any configuration if T eventually stops starting from this configuration. Indeed, such a Turing machine would solve the general halting problem by considering configurations of the form $q_i[M]C$.

9.2 Decision Problems in Group Theory

Max Dehn (1911) was among the first to work out the connection between topology and combinatorial group theory. He made explicit that answering to certain topological questions about spaces could be used to solve some general problems about group presentations. Recall that a **combinatorial presentation** $\langle S \mid R \rangle$ of a group G is defined by a set S of generators and a set R of words over[2] S, called **relations**, so that G is the quotient of the free group $F(S)$ over S by the normal closure of R in $F(S)$. Hence, the elements of G are classes of words over S where two words are in the same class if one can be transformed into the other by a sequence of insertions or removals of

1. factors $s\,s^{-1}$ with $s \in S$,
2. or words in R or their inverses.

We shall only consider *finitely presented* groups for which S and R are both finite. Most computational results nonetheless apply to *recursively presented* groups whose set of relations are recursively enumerable.

[2] By a word over S we always mean a finite sequence of elements in $S \cup S^{-1}$, where the elements of S^{-1} should be thought of as the inverses of the elements in S.

Tietze Tranformations

Clearly, a group has (infinitely) many presentations. One can indeed replace a presentation $\langle S \mid R\rangle$ by applying the following **Tietze transformations** or their inverses to obtain presentations of the same group.

T1: Add a relation which is a consequence of R.

T2: Add a new generator s with a new relation $s\,w$, where w is any word over S.

It is quite remarkable that presentations of the same group are always related by such transformations.

Theorem 9.2.1. *Two finite presentations represent the same group if and only if one can be obtained from the other by a finite sequence of Tietze transformations and their inverses.*

PROOF. Let $\langle S \mid R\rangle$ and $\langle S' \mid R'\rangle$ be two presentations of the same group. In other words, there is an isomorphism $\langle S' \mid R'\rangle \cong \langle S \mid R\rangle$. The image of any generator $s' \in S'$ under this isomorphism can be expressed as a word $s'(S)$ over S. Remark that the relations in R' are consequences of R and the relations $\{s' \cdot (s'(S))^{-1}\}_{s' \in S'}$ expressing each generator s' in terms of S (why?). For each generator $s \in S$, we define $s(S')$ analogously. We have,

$$\begin{aligned}
\langle S \mid R\rangle &\cong \langle S \cup S' \mid R \cup \{s' \cdot (s'(S))^{-1}\}_{s' \in S'}\rangle \quad \text{by repeated applications of } T_2\\
&\cong \langle S \cup S' \mid R \cup R' \cup \{s' \cdot (s'(S))^{-1}\}_{s' \in S'}\rangle \quad \text{by repeated applications of } T_1\\
&\cong \langle S \cup S' \mid R \cup R' \cup \{s' \cdot (s'(S))^{-1}\}_{s' \in S'}, \{s \cdot (s(S'))^{-1}\}_{s \in S}\rangle \quad \text{by repeated applications of } T_1
\end{aligned}$$

This last presentation is symmetric in prime and unprime symbols and could thus have been derived from $\langle S' \mid R'\rangle$. □

Exercise 9.2.2. By a consequence of R it is meant a word r on S representing an element of the normal closure of R. Show that r is freely equivalent (*i.e.*, inserting or removing $s\,s^{-1}$ or $s^{-1}s$ factors) to a word of the form

$$\prod_{j=1}^{k} g_j\, r_j^{\varepsilon_j}\, g_j^{-1},$$

where the g_j are words over S, $r_j \in R$, and $\varepsilon_j \in \{-1, 1\}$.

Exercise 9.2.3. Show that the Tietze transformations T_1 and T_2 indeed produce isomorphic groups. In other words, show that:

$$\langle S \mid R\rangle \cong \langle S \mid R \cup \{r\}\rangle \cong \langle S \cup \{s\} \mid R \cup \{s\,w\}\rangle,$$

where r is a consequence of R.

Dehn's Problems

Dehn identified three fundamental algorithmic problems [Sti87]. Let $G = \langle S \mid R \rangle$ be a finitely presented group.

- **The word problem:** decide if a word over S represents the identity in G.
- **The conjugacy problem:** decide if two words over S represent conjugate elements in G.
- **The isomorphism problem:** decide if two combinatorial presentations represent isomorphic groups.

In the late 1940's Markov and Post independently proved that the word problem in semi-groups is unsolvable. The main idea is to encode the transition of a Turing machine as relations in a semi-group. In the end the halting problem becomes equivalent to the word problem in the constructed semi-group. The unsolvability of the word problem for groups is based on similar ideas but singularly more complex. It was eventually shown by P. S. Novikov in 1955 and almost at the same time by Boone. The original article by Novikov was 143 pages long. Thanks to the HNN construction introduced by Higman, Neeumann and Neumann in 1949, Boone (1959) and Britton (1963) succeeded to reduce the proof to approximately 10 pages.

Theorem 9.2.4 (Novikov, Boone)**.** *There exists a group for which the word problem is unsolvable. In particular, the word problem for groups (given a group* and *a word as input) is unsolvable.*

The simplest example of a group with unsolvable word problem has 4 generators and 12 relations, see Borisov [Bor69]. Since the word problem is a particular case of the conjugacy problem, we immediately infer that

Corollary 9.2.5. *The conjugacy problem for groups is unsolvable.*

The **generalized word problem** is to decide if a word over the generators S of a presentation P belongs to some subgroup of P specified by a set of generators given as words over S.

Theorem 9.2.6. *The generalized word problem is unsolvable.*

Theorem 9.2.7 (Adyan 1957, Rabin 1958)**.** *The isomorphism problem for groups is unsolvable.*

A **Markov property** for groups is one that is satisfied by at least one group with finite presentation and such that there exists a group H with finite presentation such that any group including H as a subgroup does not satisfy the property. Being the trivial group, or being Abelian are Markov properties (why?). Being the fundamental group of a 3-manifold is also a Markov property because there exist finitely presentable groups which cannot appear as subgroups of 3–manifold groups.

Theorem 9.2.8 (Adyan, Rabin)**.** *If P is a Markov property, then the problem of deciding if a finite presentation satisfies P is unsolvable.*

While those negative results assert that the basic decision problems in group theory are unsolvable *in general*, there are positive results for specific classes of groups. For instance, as we saw in a previous lecture, the word and conjugacy problems are solvable for surface groups. It results from the classification of surfaces that the isomorphism problem is also solvable for surface groups. A much stronger result claims that those problems are solvable for the class of fundamental groups of closed, orientable 3–manifolds. However, none of those groups are algorithmically recognizable. Indeed, the trivial group occurs as the fundamental group of a surface group and of a closed, orientable 3–manifold group. The recognition of such groups would thus allow to decide whether a given finite presentation describes the trivial group, in contradiction with Theorem 9.2.7. See also the survey on decision problems for 3–manifolds by Aschenbrenner, Friedl and Wilton [AFW15b] for more details.

We postpone the proof of the undecidability of the word problem to Section 9.4. In the next section, we shall deduce the undecidability of topological problems from the above negative results in group theory.

9.3 Decision Problems in Topology

9.3.1 The Contractibility and Transformation Problems

Given a closed path in a simplicial complex, the **contractibility problem** is to decide if the path can be deformed continuously to a point in the complex. Likewise, given two closed path in a simplicial complex, the **transformation problem** is to decide if the paths can be deformed continuously one into the other in the complex. These are extensions of the corresponding problems we saw in the lecture on the homotopy test for surfaces.

Proposition 9.3.1. *The word and conjugacy problems respectively reduce to the contractibility and transformation problems in 2-complexes.*

The proof uses a simple construction that associates a two dimensional complex $\mathscr{C}(\langle S \mid R\rangle)$ with every group presentation $\langle S \mid R\rangle$. The complex is built from a bouquet of circles, one for each generator in S, and a set of disks, one for each non-trivial[3] relation $r \in R$. If $r = s_1^{\varepsilon_1} \cdots s_k^{\varepsilon_k}$, the boundary circle of the corresponding disk is subdivided into k subarcs and glued along the bouquet of circles in such a way that the ith arc is mapped onto the circle corresponding to generator s_i traversed in the same ($\varepsilon_i = 1$) or opposite ($\varepsilon_i = -1$) direction. By a repeated application of the Seifert–van Kampen theorem, the fundamental group of the resulting two dimensional complex is isomorphic to $\langle S \mid R\rangle$:

$$\pi_1(\mathscr{C}(\langle S \mid R\rangle)) \cong \langle S \mid R\rangle.$$

[3] For each trivial relation "1" we may also attach a sphere to the vertex of the bouquet. See the construction of Section 9.3.2.

Note that the bouquet of circles can be seen as a graph with one vertex and with one loop edge per generator. This graph is the **1-skeleton** of $\mathscr{C}(\langle S \mid R\rangle)$.

PROOF OF PROPOSITION 9.3.1. Given a word $w = s_1^{\varepsilon_1} \cdots s_k^{\varepsilon_k}$ on the generators of a presentation $\langle S \mid R\rangle$, we consider the closed path ℓ_w of length k whose ith edge is the loop edge of the 1-skeleton of $\mathscr{C}(\langle S \mid R\rangle)$ corresponding to s_i, traversed in the same ($\varepsilon_i = 1$) or opposite ($\varepsilon_i = -1$) direction. The homotopy class of ℓ_w in $\mathscr{C}(\langle S \mid R\rangle)$ is the class of w in $\langle S \mid R\rangle$, so that w represents the identity in $\langle S \mid R\rangle$ if and only if ℓ_w is contractible. Namely, the word problem for w in $\langle S \mid R\rangle$ reduces to the contractibility problem for ℓ_w in $\mathscr{C}(\langle S \mid R\rangle)$. Now, given two words u and v and their corresponding closed paths ℓ_u and ℓ_w in $\mathscr{C}(\langle S \mid R\rangle)$ we saw in the lecture on the homotopy test that ℓ_u and ℓ_w are (freely) homotopic if and only if their homotopy classes are conjugates in the fundamental group of $\mathscr{C}(\langle S \mid R\rangle)$. It follows that the conjugacy problem for u and v is equivalent to the transformation problem for ℓ_u and ℓ_w. □

Exercise 9.3.2. A 2-complex can be described as a graph, allowing loop and multiple edges, and a collection of polygons, allowing monogons and bigons, such that the boundary of each polygon is attached to a closed path in the graph. Each side of the boundary should be attached to a single edge, but the closed path need not be simple.

The barycentric subdivision of such a 2-complex is obtained by first inserting a vertex in the middle of each edge in the graph and in the middle of each side of the polygons, then triangulating each polygon by inserting a vertex at the center and joining this vertex to the boundary vertices (including the new ones) with new edges. Show that three barycentric subdivisions of $\mathscr{C}(\langle S \mid R\rangle)$ always suffice to obtain a simplicial complex.

Corollary 9.3.3. *There exists a 2-dimensional complex for which the contractibility problem is unsolvable. In particular, the contractibility problem is unsolvable for 2-complexes. The same is true for the transformation problem.*

PROOF. This follows directly from Theorem 9.2.4 and the previous Proposition 9.3.1. □

In fact, there exists a 2-dimensional complex and a closed path in this complex such that the contractibility of the path cannot be decided! The proof relies on the theory of Diophantine equations. In the famous list of 23 problems published in 1900 by Hilbert, the tenth problem asks for an algorithm to decide if a multivariable polynomial equation with integer coefficients has a solution in integers. Such equations are said **Diophantine** when one is indeed looking for integral solutions. In 1970, Matiyasevich succeeded to prove that Hilbert tenth problem is unsolvable by showing that any semi-decidable set of natural numbers is Diophantine, *i.e.*, has the form

$$\{n \in \mathbb{N} \mid \exists (n_1, \ldots, n_k) \in \mathbb{Z}^k : p(n, n_1, \ldots, n_k) = 0\}$$

for some polynomial p with integer coefficients in $k+1$ variables. Now, the set of statements in any formal system with recursively enumerable description (axioms and inference rules) can be numbered so that the theorems form a semi-decidable subset.

By Gödel first incompleteness theorem, such a system, assuming it can express basic facts about natural numbers, has a statement that can neither be proved or disprove in the system (such as stating its own consistency, which cannot be proved by Gödel second incompleteness theorem). If n is the number of an undecidable statement and p is the Diophantine equation for the set of theorems, then it cannot be decided if $p(n,\cdot)$ has a solution. (See [Jon82, CM14] for explicit constructions.) More precisely, it cannot be proved that $p(n,\cdot)$ has no solution (if $p(n,\cdot)$ had a solution, this solution would provide its own proof). Hence, considering a Turing machine M that looks for a solution of $p(n,\cdot)$, we cannot prove that the machine runs indefinitely given $p(n,\cdot)$ as input. In Section 9.4 we shall construct, for every Turing machine and every input, a group presentation P with a word w in its generators such that the machine eventually halts after being given the input if and only if w represents the identity in P. This provides a 2-complex $\mathscr{C}(P)$ and a closed path corresponding to (an encoding of) $p(n,\cdot)$ for which we cannot prove that the path is non-contractible.

Remark 9.3.4. The results in this section extend to four dimensional manifolds since any finitely presented group can be realized as the fundamental group of a 4-manifolds that can effectively be computed (Dehn 1910).

9.3.2 The Homeomorphism Problem

The **homeomorphism problem** is to decide if two given combinatorial spaces, say simplicial complexes, are homeomorphic. Since we know that the isomorphism problem is unsolvable (Theorem 9.2.7), it is tempting to use the 2-complex $\mathscr{C}(P)$ associated to a group presentation P to reduce the isomorphism problem to the homeomorphism problem and conclude that this last one is also unsolvable. Indeed, if the complexes $\mathscr{C}(P)$ and $\mathscr{C}(Q)$ corresponding to the group presentations P and Q are homeomorphic, then their fundamental groups, hence P and Q, are isomorphic. However, different presentations of the same group may lead to non-homeomorphic 2-complexes so that we cannot conclude that the group are distinct when the corresponding 2-complexes are not homeomorphic. As a simple example consider the presentations $\langle \{s\} \mid \{s\}\rangle$, $\langle \{s\} \mid \{s,s\}\rangle$, and $\langle \{s\} \mid \{s,1\}\rangle$ of the trivial group. The corresponding 2-complexes are respectively homeomorphic to a disk, a sphere, and a sphere attached to a disk through a point. In order to prove the unsolvability of the homeomorphism problem one needs a presentation-invariant construction of a complex whose fundamental group is the given group. This was eventually achieved by Markov, using four dimensional manifolds rather than 2-complexes. Markov's proof is based on a Seifert and Threlfall construction (1934) using manifold surgery. Following Stillwell, we shall rely on a construction of Boone, Haken and Poénaru (1968).

Theorem 9.3.5 (Markov, 1958)**.** *The homeomorphism problem is unsolvable for manifolds of dimension* 4 *or larger.*

PROOF. Given two presentations P and Q we shall construct 4-manifold complexes $\mathscr{C}'(P)$ and $\mathscr{C}'(Q)$ such that $P \cong Q$ if and only if $\mathscr{C}'(P)$ and $\mathscr{C}'(Q)$ are homeomorphic. Since isomorphic presentations are related by Tietze transformations (Theorem 9.2.1) a solution is to provide a construction whose homeomorphism type is invariant by

Tietze transformations. The above examples show that this is not the case for $\mathscr{C}(P)$. It turns out that the extra sphere arising from the trivial relation in the examples is essentially the only obstruction to an invariant construction. To overcome this problem Boone et al. introduce three modifications.

1. If P has p generators and m relations and Q has q generators and n relations, first add $p+n+1$ trivial relations (1) to P and $q+m+1$ trivial relations to Q. Denote by $P\star(p+n+1)$ and $Q\star(q+m+1)$ the resulting presentations.

2. Replace the 2-complexes $\mathscr{C}(P)$ and $\mathscr{C}(Q)$ by their thickening in $\mathbb{R}^5$. First note that any 2-complex $\mathscr{C}$ can be triangulated (see Exercise 9.3.2) and that any such triangulation has a piecewise linear (PL) embedding in $\mathbb{R}^5$. For $\varepsilon>0$, let $\mathscr{C}^\varepsilon$ be the set of points at distance at most ε from $\mathscr{C}$ in $\mathbb{R}^5$. When ε is small enough $\mathscr{C}^\varepsilon$ deform retracts[4] onto $\mathscr{C}$, hence has the same fundamental group as $\mathscr{C}$. Moreover, $\mathscr{C}^\varepsilon$ can be triangulated and such a triangulation can be computed from $\mathscr{C}$. We set $\mathscr{C}'(P)$ to the boundary of the 5-manifold $\mathscr{C}^\varepsilon(P\star(p+n+1))$ and $\mathscr{C}'(Q)$ to the boundary of $\mathscr{C}^\varepsilon(Q\star(q+m+1))$.

3. In order to prove the invariance by Tietze transformations, replace the addition of a consequence relation (T_1) by four transformations T_{11}, T_{12}, T_{13} and T_{14}:

   T_{11} : replace a relation r by $s\,s^{-1}\,r$ or $s^{-1}\,s\,r$ for some generator s,

   T_{12} : replace a relation $u\,v\,w$ by a cyclic permutation $v\,w\,u$,

   T_{13} : replace a relation r by r^{-1},

   T_{14} : replace r by $r\,r'$ where r, r' are the ith and jth relations, $i\neq j$.

   Hence, transformation T_{1i} replaces a relation rather than adding a new one. It clearly produces isomorphic presentations (prove it!).

Claim 1. Let $P=\langle S\mid R\rangle$ and $P'=\langle S\mid R\cup\{r\}\rangle$, where r is a consequence of R. Then $P\star 2$ may be converted to $P'\star 1$ using transformations $T_{11},\dots,T_{14}$ and T_2 and their inverses.

PROOF. By Exercise 9.2.2, we may write $r=\prod_{j=1}^{k} g_j\,r_j^{\varepsilon_j}\,g_j^{-1}$. By a combination of $T_{11},\dots,T_{14}$ and their inverses, we can transform the second of the two extra relations in $P\star 2$ into $g_j\,r_j^{\varepsilon_j}\,g_j^{-1}$. We can then use transformation T_{14} to accumulate such factors in the first extra relation, resetting each time the second extra relation to 1 by the reverse transformations used to get $g_j\,r_j^{\varepsilon_j}\,g_j^{-1}$. The details are left to the reader. □

Claim 2. If P and Q are isomorphic then we can transform $P\star(n+m+1)$ into $Q\star(n+m+1)$ using a sequence of transformations $T_{11},\dots,T_{14}$ and T_2 and their inverses.

PROOF. Let $P=\langle S\mid R\rangle$ and $Q=\langle S'\mid R'\rangle$. Using the notations in the proof of Theorem 9.2.1 we first transform $P\star(p+n+1)$ into $\langle S\cup S'\mid R\cup\{s'\cdot(s'(S))^{-1}\}_{s'\in S'}\rangle\star(p+n+1)$ by repeated applications of T_2. We further mimic the proof of Theorem 9.2.1 using

[4] A **retraction** is a continuous map from a topological space onto a subspace whose restriction to the subspace is the identity map. A **deformation retraction** is a homotopy between the identity map and a retraction.

combinations of transformations $T_{11},\dots,T_{14}$ in place of T_1. We obtain this way the isomorphic presentation $\langle S\cup S' \mid R\cup R'\cup\{s'\cdot(s'(S))^{-1}\}_{s'\in S'},\{s\cdot(s(S'))^{-1}\}_{s\in S}\rangle \star 1$ which is symmetric in prime and unprime symbols and could thus have been derived from $Q\star(q+m+1)$. □

Claim 3. If presentation P_2 results from presentation P_1 by a transformation $T_{11},\dots,T_{14}$ or T_2, then $\mathscr{C}^\varepsilon(P_2)$ is homeomorphic to $\mathscr{C}^\varepsilon(P_1)$.

PROOF. The claim is trivial for transformations T_{12} and T_{13} since the 2-complexes $\mathscr{C}(P_1)$ and $\mathscr{C}(P_2)$ are the same in those cases. Consider now the transformation T_{11} applied to P_1. It replaces one of its relations r by $s\,s^{-1}\,r$ (or $s^{-1}\,s\,r$). Let P_0 be P_1 minus the relation r, which is also P_2 minus the relation $s\,s^{-1}\,r$. The 2-complex $\mathscr{C}(P_1)$ is obtained from $\mathscr{C}(P_0)$ by attaching a disk D to the closed curve corresponding to r in the 1-skeleton of $\mathscr{C}(P_0)$. Disk D intersects the thickening $\mathscr{C}^\varepsilon(P_0)$ in a simple closed curve ℓ_1 which cuts D into a smaller disk D_1 outside $\mathscr{C}^\varepsilon(P_0)$. So, $\mathscr{C}^\varepsilon(P_1)$ is the union of $\mathscr{C}^\varepsilon(P_0)$ and the thickening D_1^ε of D_1. Likewise, $\mathscr{C}^\varepsilon(P_2)$ is the union of $\mathscr{C}^\varepsilon(P_0)$ and the thickening D_2^ε of a disk D_2 that intersects $\mathscr{C}^\varepsilon(P_0)$ in a simple closed curve ℓ_2. Now, ℓ_1 and ℓ_2 differ by a thin "tongue" close to the path $s\,s^{-1}$. Hence, there is a homeomorphism (in fact an ambient isotopy) of $\mathscr{C}^\varepsilon(P_0)$ sending ℓ_2 to ℓ_1. We can extend this homeomorphism to an homeomorphism between $\mathscr{C}^\varepsilon(P_2)=\mathscr{C}^\varepsilon(P_0)\cup D_2^\varepsilon$ and $\mathscr{C}^\varepsilon(P_1)=\mathscr{C}^\varepsilon(P_0)\cup D_1^\varepsilon$. Similar constructions hold for the last two transformations T_{14} and T_2. See [Sti93, Sec. 9.4.4] for the details. □

We are now ready to prove that $\mathscr{C}'(P)$ and $\mathscr{C}'(Q)$ are homeomorphic if and only if P and Q are isomorphic. Recall that the fundamental group of $\mathscr{C}^\varepsilon(P\star(p+n+1))$ is $P\star(p+n+1)\cong P$. Since $\mathscr{C}^\varepsilon(P\star(p+n+1))$ is a 5-manifold, removing its 2-dimensional core $\mathscr{C}(P\star(p+n+1))$ does not change its fundamental group. Moreover, since $\mathscr{C}^\varepsilon(P\star(p+n+1))\setminus\mathscr{C}(P\star(p+n+1))$ deform retracts onto the boundary of $\mathscr{C}^\varepsilon(P\star(p+n+1))$ they also have the same fundamental group. We conclude that $\pi_1(\mathscr{C}'(P))\cong P$. Likewise, $\pi_1(\mathscr{C}'(Q))\cong Q$. It follows that $\mathscr{C}'(P)$ and $\mathscr{C}'(Q)$ cannot be homeomorphic if P and Q are not homeomorphic.

Suppose now that P and Q are isomorphic. According to Claim 2, $P\star(p+n+1)$ can be converted to $Q\star(q+m+1)$ using a sequence of transformations $T_{11},\dots,T_{14}$ and T_2 and their inverses. Following Claim 3, $\mathscr{C}^\varepsilon(P\star(p+n+1))$ and $\mathscr{C}^\varepsilon(Q\star(q+m+1))$ are homeomorphic and so are their boundaries $\mathscr{C}'(P)$ and $\mathscr{C}'(Q)$. □

Exercise 9.3.6. Prove that any finite 2-dimensional simplicial complex has a PL embedding in $\mathbb{R}^5$.

Exercise 9.3.7. Provide the details in the proof of the above Claim 1.

Quite surprisingly, while there is no algorithm to decide whether two 2-complexes have isomorphic fundamental groups, the homeomorphism problem for 2-complexes is solvable! This results from the existence of a normal form for 2-complexes due to Whittlesey [Whi58, Whi60]. This normal form easily leads to an equivalence between the homeomorphism problem for 2-complexes and the graph isomorphism problem [STP94, DWW00]. The homeomorphism problem is also solvable for closed, oriented, triangulated 3-manifolds as recently proved by Kuperberg [Kup15b]. The proof relies on the geometrization theorem conjectured by Thurston and proved by Perelman. This geometrization theorem provides a canonical decomposition of 3-manifolds into elementary pieces that can be algorithmically recognized.

9.4 Proof of the Undecidability of the Group Problems

In this Section we give a complete proof of Theorems 9.2.4, 9.2.6 and 9.2.7. We follow the proof by Stillwell [Sti82, Sti93][5]. A first step is two replace Turing machines by the $\mathbb{Z}^2$-machine formalism.

9.4.1 $\mathbb{Z}^2$-Machines

We can interpret a Turing machine $M = (\mathscr{A}, \mathscr{Q}, \mathscr{T})$ as a set of transformations over $\mathbb{Z}^2$. To this end we associate with every letter and state of M a distinct digit in base β between 0 and $\beta - 1$, where $\beta = |\mathscr{A}| + |\mathscr{Q}|$. For a word w in $(\mathscr{A} \cup \mathscr{Q})^*$, let $\mathscr{B}(w)$ be the integer in base β whose digits are associated with the letters and states of w, in the same order. We encode a configuration uqv of M as a couple $(\mathscr{B}(uq), \mathscr{B}(\bar{v}))$ of integers, where $\bar{v} = \overline{v_1 v_2 \dots v_k} = v_k \dots v_2 v_1$. Every transition of M may be interpreted as a partial transformation over $\mathbb{Z}^2$. Precisely, we associate with every transition $aqbpL$ the **l-transformations**:

$$(\beta^2 U + \mathscr{B}(cq), \beta V + \mathscr{B}(a)) \xrightarrow{l} (\beta U + \mathscr{B}(p), \beta^2 V + \mathscr{B}(bc))$$

corresponding to the transitions $\mathscr{B}^{-1}(U)cqa\overline{\mathscr{B}^{-1}(V)} \mapsto \mathscr{B}^{-1}(U)pcb\overline{\mathscr{B}^{-1}(V)}$. Those transformations can be written as

$$(\beta^2 U + A_l, \beta V + B_l) \xrightarrow{l} (\beta U + C_l, \beta^2 V + D_l)$$

for some appropriate numbers A_l, B_l, C_l, D_l. Those four numbers determine the l-transformation. Note that a single transition gives rise to a number $|\mathscr{A}|$ of l-transformations, one for each $c \in \mathscr{A}$. Similarly, every transition $aqbpR$ is associated the **r-transformations**:

$$(\beta U + \mathscr{B}(q), \beta^2 V + \mathscr{B}(ca)) \xrightarrow{r} (\beta^2 U + \mathscr{B}(bp), \beta V + \mathscr{B}(c))$$

which write

$$(\beta U + A_r, \beta^2 V + B_r) \xrightarrow{r} (\beta^2 U + C_r, \beta V + D_r)$$

for appropriate A_r, B_r, C_r, D_r.

For numbers X, Y, X', Y', we write $(X, Y) \xrightarrow{s} (X', Y')$ if (X', Y') is the result of an s-transformation, $s \in \{l, r\}$, applied to (X, Y). More generally, we write

$$(X, Y) \xrightarrow{*} (X', Y')$$

if (X', Y') is obtained from (X, Y) by applying a finite sequence of transformations. Hence, M changes from a configuration to another one by a sequence of transitions if and only if $(X, Y) \xrightarrow{*} (X', Y')$ for the corresponding $\mathbb{Z}^2$-couples. We finally write

$$(X, Y) \xleftrightarrow{*} (X', Y')$$

if there exist $\mathbb{Z}^2$-couples $(X, Y) = (X_0, Y_0), (X_1, Y_1), \dots, (X_n, Y_n) = (X', Y')$ such that, for $0 \le i < n$, either $(X_i, Y_i) \xrightarrow{s_i} (X_{i+1}, Y_{i+1})$ or $(X_{i+1}, Y_{i+1}) \xrightarrow{s_i} (X_i, Y_i)$, where $s_i \in \{l, r\}$.

[5] Another interesting but incomplete presentation is proposed by Andrews [And05].

We shall prove that the halting problem for Turing machines is Turing reducible to the generalized word problem. For this, we consider the $\mathbb{Z}^2$-machine Z corresponding to an arbitrary Turing machine. We then construct a group K_Z and a 1-1 map $p : \mathbb{Z}^2 \to K_Z$ so that the statement

$$Z\text{, starting from some } (u,v) \in \mathbb{Z}^2\text{, eventually stops}$$

is equivalent to $p(u,v)$ belonging to a certain subgroup of K_Z. We start recalling fundamental constructions in group theory.

9.4.2 Useful Constructs in Combinatorial Group Theory

Free Groups and Free Products

Recall that a free group over a set S is the group $F(S) = \langle S \mid - \rangle$ of words over S modulo the insertion of trivial relations $s\,s^{-1}$ and $s^{-1}s$, $s \in S$.

A relation between elements of a group is any product of those elements and their inverses which is the identity in the group. A relation is **reduced** if it does not contain two inverse consecutive factors. A subgroup H of a group G is free if H is isomorphic to a free group. A subset $S \subset G$ is a **free basis** for the subgroup it generates if there is no non-trivial reduced relations between the elements of S. In this case, the subgroup generated by S is a free subgroup isomorphic to $F(S)$.

The **free product** of two groups with presentations $\langle S \mid R \rangle$ and $\langle S' \mid R' \rangle$ is the group $\langle S \mid R \rangle * \langle S' \mid R' \rangle := \langle S \cup S' \mid R \cup R' \rangle$ (here, S, S' must be considered as disjoint sets even when $\langle S \mid R \rangle \cong \langle S' \mid R' \rangle$). The free product only depends on the group factors and not on the used presentations[6]. The **normal form theorem for free products** says that any non-trivial element of $G * H$ may be uniquely written as an alternating product of non-trivial elements of G and non-trivial elements of H. In particular, G and H embeds as subgroups of $G * H$.

HNN Extension and Britton's Lemma

Given a group $G = \langle S \mid R \rangle$ and an isomorphism $\varphi : A \to B$ between two subgroups A and B of G, Graham Higman, Bernhard Neumann et Hanna Neumann (1949) established the existence of a group $G*_\varphi$ containing G as a subgroup and such that $\varphi : A \to B$ becomes an inner automorphism (A and B are conjugate subgroups) in $G*_\varphi$. More precisely,

Definition 9.4.1. The **HNN extension** of G relatively to φ is the group

$$G*_\varphi := \langle S \cup \{t\} \mid R \cup \{\varphi(a) = t^{-1} a\, t\}_{a \in A} \rangle$$

where t is a new generator qualified as **stable**.

An essential property of HNN extensions is the existence of some kind of normal forms resulting from the following Britton's lemma.

[6] Free products can be defined by a universal property.

Lemma 9.4.2 (Britton, 1963)**.** *If a product $g_0 t^{\epsilon_1} g_1 t^{\epsilon_2} \dots t^{\epsilon_n} g_n$ represents the identity in $G*_\varphi$, where $g_i \in G$ and $\epsilon_i \in \{-1,1\}$, $\forall i \in [0,n]$, then either $n=0$ and $g_0 =_G 1$, or for some $i \in [1,n-1]$ we have*

- *either $\epsilon_i = -1, \epsilon_{i+1} = 1$ and $g_i \in A$,*
- *or $\epsilon_i = 1, \epsilon_{i+1} = -1$ and $g_i \in B$.*

Corollary 9.4.3 (Normal form for HNN extentions)**.** *Every element of $G*_\varphi$ has a unique expression as $g_0 t^{\epsilon_1} g_1 t^{\epsilon_2} \dots t^{\epsilon_n} g_n$, where*

- *for $0 < i < n$, $g_i = 1$ implies $\epsilon_i = \epsilon_{i+1}$,*
- *$\epsilon_i = 1$ implies $g_i \in A$,*
- *$\epsilon_i = -1$ implies $g_i \in B$.*

Here, uniqueness applies to n, the ϵ_i, and the right coset representatives of the g_i modulo A and B respectively.

9.4.3 Undecidability of the Generalized Word Problem

Let

$$K = \langle x, y, z \mid [x,y] \rangle \quad \text{and} \quad p : \begin{array}{ccc} \mathbb{Z}^2 & \longrightarrow & K \\ (u,v) & \longmapsto & (x^u y^v)^{-1} z x^u y^v \end{array}$$

Note that $K \cong \langle x, y \mid [x,y] \rangle * \langle z \mid - \rangle \cong \mathbb{Z}^2 * \mathbb{Z}$

Lemma 9.4.4. *The image of $\mathbb{Z}^2$ under the map p forms a free basis of a free subgroup of K. In particular, p is one-to-one.*

PROOF. Let $w = p(u_1,v_1)^{j_1} \cdot p(u_2,v_2)^{j_2} \dots p(u_n,v_n)^{j_n}$ be a *reduced* product of $p(u,v)$ factors, *i.e.*, such that $(u_i,v_i) \neq (u_{i+1},v_{i+1})$ and $j_i \neq 0$. Substituting the $p(u_i,v_i)$ with their values and using that x and y commute in K, we get

$$w =_K x^{-u_1} y^{-v_1} z^{j_1} x^{u_1-u_2} y^{v_1-v_2} z^{j_2} \dots x^{u_{n-1}-u_n} y^{v_{n-1}-v_n} z^{j_n} x^{u_n} y^{v_n}.$$

From the normal form theorem of free products, if w is the identity in K, then it contains a factor $x^{u_i-u_{i+1}} y^{v_i-v_{i+1}}$ which is 1 in $\langle x, y \mid [x,y] \rangle$. However this is in contradiction with the hypothesis that $(u_i,v_i) \neq (u_{i+1},v_{i+1})$. It follows that the $p(u_i,v_i)$ constitute a free basis. □

With every l-transformation, we associate a morphism

$$\phi_l : < x^{\beta^2}, y^\beta, p(A_l,B_l) > \rightarrow < x^\beta, y^{\beta^2}, p(C_l,D_l) >$$

between the two subgroups of K respectively generated by $x^{\beta^2}, y^\beta, p(A_l,B_l)$ and $x^\beta, y^{\beta^2}, p(C_l,D_l)$. This morphism is defined by $x^{\beta^2} \mapsto x^\beta, y^\beta \mapsto y^{\beta^2}$ and $p(A_l,B_l) \mapsto$

$p(C_l, D_l)$. That this indeed defines a morphism is not obvious, see the next lemma. We similarly associate with every r-transformation the morphism

$$\phi_l :< x^\beta, y^{\beta^2}, p(A_r, B_r) > \longrightarrow < x^{\beta^2}, y^\beta, p(C_r, D_r) >$$

defined by $x^\beta \mapsto x^{\beta^2}, y^{\beta^2} \mapsto y^\beta, p(A_r, B_r) \mapsto p(C_r, D_r)$.

Lemma 9.4.5. *The maps ϕ_l and ϕ_r are well-defined isomorphisms.*

PROOF. Let ρ_l be the (inner) automorphism acting by conjugation by $x^{-A_l} y^{-B_l}$. This morphism sends $< x^{\beta^2}, y^\beta, p(A_l, B_l) >$ isomorphically onto $< x^{\beta^2}, y^\beta, z >$. Similarly, we have an inner automorphism θ_l sending $< x^\beta, y^{\beta^2}, p(C_l, D_l) >$ onto $< x^\beta, y^{\beta^2}, z >$. Now, we just need to show that $\theta_l \circ \phi_l \circ \rho_l^{-1}$ exists and is an isomorphism. To see this, first note that $< x^{\beta^2}, y^\beta, z >$ as a subgroup of K is equal to $< x^{\beta^2}, y^\beta > * < z >$ (show inclusion in both directions) and similarly $< x^\beta, y^{\beta^2}, z >=< x^\beta, y^{\beta^2} > * < z >$. Now, the map $x^{\beta^2} \mapsto x^\beta, y^\beta \mapsto y^{\beta^2}$ induces an isomorphism $< x^{\beta^2}, y^\beta > \longrightarrow < x^\beta, y^{\beta^2} >$ between groups isomorphic to $\mathbb{Z}^2$. The "free product" of this isomorphism with the identity over $< z >$ is an isomorphism and is precisely $\theta_l \circ \phi_l \circ \rho_l^{-1}$. The same proof holds for ϕ_r, substituting r for l. □

We can thus consider the HNN extension $K*_{\phi_l}$ of K by ϕ_l. Let t_l be the stable generator of this extension.

Lemma 9.4.6. *$(u, v) \xrightarrow{l} (u', v')$ if and only if $t_l^{-1} p(u, v) t_l = p(u', v')$ in $K*_{\phi_l}$. Likewise, $(u, v) \xrightarrow{r} (u', v')$ if and only if $t_r^{-1} p(u, v) t_r = p(u', v')$ in $K*_{\phi_r}$.*

PROOF. If $(u, v) \xrightarrow{l} (u', v')$ then for some numbers U, V : $u = \beta^2 U + A_l, v = \beta V + B_l, u' = \beta U + C_l, v' = \beta^2 V + D_l$. It easily follows that $\phi_l(p(u, v)) = p(u'v')$, whence $t_l^{-1} p(u, v) t_l = p(u', v')$ in $K*_{\phi_l}$. Conversely, suppose that $t_l^{-1} p(u, v) t_l p(u', v')^{-1} = 1$. By Britton's Lemma 9.4.2 applied to $K*_{\phi_l}$, we have $p(u, v) \in < x^{\beta^2}, y^\beta, p(A_l, B_l) >$. Hence,

$$p(u, v) = x^{\beta^2 j_1} y^{\beta j_2} p(A_l, B_l)^{j_3} x^{\beta^2 j_4} \dots p(A_l, B_l)^{j_n} \tag{9.1}$$

for some integers $j_1, j_2, \dots, j_n$. Using trivial relations and the commutation of x and y, the right-hand side of (9.1) may be written as

$$p(\beta^2 U_1 + A_l, \beta V_1 + B_l)^{j_3} \cdot p(\beta^2 U_2 + A_l, \beta V_2 + B_l)^{j_6} \dots p(\beta^2 U_k + A_l, \beta V_k + B_l)^{j_n} x^{\beta^2 p} y^{\beta q}$$

for some $U_1, V_1, U_2, V_2, \dots U_k, V_k, p, q$. Making x, y and z commute (by Abelianizing K) in this equality, we deduce that $p = q = 0$. By Lemma 9.4.4, we then conclude that the right-hand side member of (9.1) reduces to a single factor $p(\beta^2 U + A_l, \beta V + B_l)$ for which $u = \beta^2 U + A_l$ and $v = \beta V + B_l$. We thus compute in $K*_{\phi_r}$ that $t_l^{-1} p(u, v) t_l = \phi_l(p(u, v)) = p(\beta U + C_l, \beta^2 V + D_l)$. It then follows from the hypothesis $p(u', v') = t_l^{-1} p(u, v) t_l$ and Lemma 9.4.4 that $u' = \beta U + C_l$ et $v' = \beta^2 V + C_l$. In other words, $(u, v) \xrightarrow{l} (u', v')$. The case of an r-transformation may be treated the same way. □

Denote by K_Z the group obtained from K by the successive HNN extensions by the morphisms ϕ_l and ϕ_r associated with all the l and r-transformations of Z. Clearly, the resulting group does not depend on the order of successive extensions. Since a group embeds in all its extensions, the previous lemma remains valid in K_Z. Denote by $\{t_l\}$ and $\{t_r\}$ the stable generators of all the HNN extensions corresponding to the φ_l and φ_r morphisms.

Lemma 9.4.7. *$(u',v') \overset{*}{\longleftrightarrow} (u,v)$ if and only if $p(u',v') \in < p(u,v), \{t_r\}, \{t_l\} > \subset K_Z$.*

PROOF. By repeated applications of Lemma 9.4.6, if $(u',v') \overset{*}{\longleftrightarrow} (u,v)$ then there exists $w \in < \{t_l\}, \{t_r\} > \subset K_Z$ such that $p(u',v') = w^{-1} p(u,v) w$. In particular, $p(u',v') \in < p(u,v), \{t_r\}, \{t_l\} >$. Conversely, suppose that $p(u',v') \in < p(u,v), \{t_r\}, \{t_l\} >$. Hence, $p(u',v')$ may be written

$$T_0 p(u,v)^{j_1} T_1 p(u,v)^{j_2} \dots p(u,v)^{j_k} T_k \tag{9.2}$$

for some integers $j_1, j_2, \dots, j_k$ and words $T_0, T_1, \dots, T_k$ in $< \{t_r\}, \{t_l\} >$. Since the value $p(u',v')$ of this product is in K, it follows by induction on the number of HNN extensions from K to K_Z and by Britton's lemma that this product contains a factor of the form $t_s^{\pm 1} w t_s^{\mp 1}$, where w is in the domain or codomain of ϕ_s. We must have $w = p(u,v)^j$ for some $j \in \{j_1, j_2, \dots, j_k\}$, so that

$$t_s^{\pm 1} w t_s^{\mp 1} = t_s^{\pm 1} p(u,v)^j t_s^{\mp 1} = (t_s^{\pm 1} p(u,v) t_s^{\mp 1})^j = p(u'',v'')^j$$

where either $(u,v) \overset{s}{\rightarrow} (u'',v'')$ or $(u,v) \overset{s}{\leftarrow} (u'',v'')$ depending on the signs in the t_s exponents. In particular, the fact that $p(u,v)^j$ is in the (co)domain of ϕ_s implies that the s-transformation (or its inverse) corresponding to t_s applies to (u,v). Substituting $p(u'',v'')^j$ to $t_s^{\pm 1} p(u,v)^j t_s^{\mp 1}$ in (9.2) we get a new expression in terms of the T_i's, $p(u,v)$ and $p(u'',v'')$. Iterating the process we eventually obtain

$$p(u',v') = p(u_1,v_1)^{j_1} p(u_2,v_2)^{j_2} \dots p(u_k,v_k)^{j_k}$$

where $(u_i,v_i) \overset{*}{\longleftrightarrow} (u,v)$ for each i. Lemma 9.4.4 allows to conclude that the right-hand side reduces to a single $p(u_i,v_i)$ with $(u_i,v_i) = (u',v')$, so that $(u',v') \overset{*}{\longleftrightarrow} (u,v)$. □

Lemma 9.4.8. *Let $(u_0,v_0) \in \mathbb{Z}^2$ corresponds to a halting configuration of Z. Then, $(u,v) \overset{*}{\longleftrightarrow} (u_0,v_0)$ if and only if $(u,v) \overset{*}{\rightarrow} (u_0,v_0)$.*

PROOF. On the one hand, we cannot have $(u,v) \overset{s}{\leftarrow} (u_0,v_0)$ since (u_0,v_0) is halting. On the other hand, $(u,v) \overset{s}{\leftarrow} (u',v') \overset{s'}{\rightarrow} (u'',v'')$ implies $(u,v) = (u'',v'')$ since Z is determinist. We can thus assume that this pattern does not occur in $(u,v) \overset{*}{\longleftrightarrow} (u_0,v_0)$. It follows that $(u,v) \overset{*}{\rightarrow} (u_0,v_0)$. □

PROOF OF THEOREM 9.2.6. let Z be the $\mathbb{Z}^2$-machine corresponding to a universal Turing machine T. Up to a simple modification, we can assume that T has a unique halting configuration corresponding to some (u_0,v_0) for Z. It follows from Lemmas 9.4.7 and 9.4.8 that T eventually halts starting from a configuration with $\mathbb{Z}^2$ code (u,v) if and only if $p(u,v)$ belongs to the subgroup $< p(u_0,v_0), \{t_r\}, \{t_l\} >$ of K_Z. This last generalized word problem is thus unsolvable by Corollary 9.1.2. □

PROOF OF THEOREM 9.2.4. Let $H =< p(u_0, v_0), \{t_r\}, \{t_l\} > \subset K_Z$. Consider the HNN extension $L := K_Z *_{Id_H}$ and let k be the stable generator of this extension. By Britton's lemma $p(u,v)kp(u,v)^{-1}k^{-1} =_L 1$ if and only if $p(u,v) \in H$. Hence, the generalized word problem reduces to the word problem. □

PROOF OF THEOREM 9.2.7. The unsolvability of the isomorphism problem results from Theorem 9.2.8 since being isomorphic to the trivial group is a Markov property. For completeness we nonetheless provide an independent proof based on the above construction. First note that all non-trivial elements of K_Z have infinite order. This is true for $K = \langle x, y, z \mid [x, y] \rangle$ and its embedding in K_Z. Moreover, applying Britton's lemma to the powers of the normal form of an element involving a stable generator shows that such an element has also infinite order.

Let $\langle \{s_1, s_2, \ldots, s_n\} \mid R \rangle$ be the presentation of K_Z naturally obtained by the successive extensions. In particular, $\{s_1, s_2, \ldots, s_n\} = \{x, y, z\} \cup \{t_l\} \cup \{t_r\}$. For a word w in the s_i's, we consider the group with presentation

$$K_Z(w) := \langle \{s_1, s_2, \ldots, s_n\} \cup \{k_i\}_{1 \leq i \leq n} \mid R \cup \{k_i^{-1} w k_i = s_i\}_{1 \leq i \leq n} \rangle$$

We claim that w represents the identity in K_Z if and only if $K_Z(w)$ is isomorphic to the free group over n elements. Indeed, if $w =_{K_Z} 1$ then the new relations in $K_Z(w)$ becomes $1 = s_i$, whence $K_Z(w) = \langle \{k_i\}_{1 \leq i \leq n} \mid - \rangle$. Conversely, if $w \neq 1$ then w, like the s_i's, has infinite order in K_Z. It follows that $w \mapsto s_i$ defines an isomorphism of cyclic infinite groups. Hence, $K_Z(w)$ may be viewed as resulting from a sequence of HNN extensions with stable generators $\{k_i\}_{1 \leq i \leq n}$. In particular, K_Z embeds in $K_Z(w)$, implying that the word problem for $K_Z(w)$ is unsolvable. On the other hand, if $K_Z(w)$ was isomorphic to a free group then the word problem for $K_Z(w)$ would be solvable, a contradiction. Hence, $K_Z(w)$ is not isomorphic to a free group, thus proving the claim. It follows that this instance of the isomorphism problem reduces to an unsolvable word problem. □

Bibliography

[ACC16] Aaron Adcock, Erik Carlsson, and Gunnar Carlsson. The ring of algebraic functions on persistence bar codes. *Homology, Homotopy and Applications*, 18(1):381–402, 2016.

[ACNS82] Miklós Ajtai, Vašek Chvátal, Monroe M Newborn, and Endre Szemerédi. Crossing-free subgraphs. *North-Holland Mathematics Studies*, 60:9–12, 1982.

[AFW15a] Matthias Aschenbrenner, Stefan Friedl, and Henry Wilton. Decision problems for 3–manifolds and their fundamental groups. *Geometry & Topology Monographs*, 19(1):201–236, 2015.

[AFW15b] Matthias Aschenbrenner, Stefan Friedl, and Henry Wilton. Decision problems for 3–manifolds and their fundamental groups. *Geometry & Topology Monographs*, 19(1):201–236, 2015.

[AHT06] Ian Agol, Joel Hass, and William Thurston. The computational complexity of knot genus and spanning area. *Transactions of the American Mathematical Society*, 358:3821–3850, 2006.

[And05] Uri Andrews. Undecidable problems in topology. `https://math.berkeley.edu/~hutching/teach/215b-2005/andrews.ps`, May 2005.

[Ban74] Thomas Banchoff. Triple points and singularities of projections of smoothly immersed surfaces. *Proceedings of the American Mathematical Society*, 46(3):402–406, 1974.

[BCFN16] Glencora Borradaile, Erin Wolf Chambers, Kyle Fox, and Amir Nayyeri. Minimum cycle and homology bases of surface embedded graphs. In *32nd International Symposium on Computational Geometry (SoCG 2016)*, volume 51. Schloss Dagstuhl-Leibniz-Zentrum fur Informatik, 2016.

[Bec12] J. Beckham. Analytics reveal 13 new basketball positions. *Wired*, 2012.

[Bor69] V. V. Borisov. Simple examples of groups with unsolvable word problem. *Mathematical Notes*, 6(5):768–775, 1969.

[Bra09] Ulrik Brandes. The left-right planarity test. http://www.inf.uni-konstanz.de/algo/publications/b-lrpt-sub.pdf, 2009.

[BS14] Peter Bubenik and Jonathan A Scott. Categorification of persistent homology. *Discrete & Computational Geometry*, 51(3):600–627, 2014.

[BZ85] Gerhard Burde and Heiner Zieschang. *Knots*. De Gruyter, 1985.

[CB15] William Crawley-Boevey. Decomposition of pointwise finite-dimensional persistence modules. *Journal of Algebra and Its Applications*, 14(05):1550066, 2015.

[CCE13] Sergio Cabello, Erin W. Chambers, and Jeff Erickson. Multiple-source shortest paths in embedded graphs. *SIAM Journal on Computing*, 42(4):1542–1571, 2013.

[CCSG+09] F. Chazal, D. Cohen-Steiner, M. Glisse, L. Guibas, and S. Oudot. Proximity of persistence modules and their diagrams. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*, pages 237–246. ACM, 2009.

[CDSGO12] Frédéric Chazal, Vin De Silva, Marc Glisse, and Steve Oudot. The structure and stability of persistence modules. arXiv preprint arXiv:1207.3674, 2012.

[CdVPV03] Éric Colin de Verdière, Michel Pocchiola, and Gert Vegter. Tutte's barycenter method applied to isotopies. *Computational Geometry: Theory and Applications*, 26(1):81–97, 2003.

[CG83] John H. Conway and Cameron McA. Gordon. Knots and links in spatial graphs. *Journal of Graph Theory*, 7(4):445–453, 1983.

[CLRS02] T. Cormen, C. Leiserson, R. Rivest, and C. Stein. *Introduction to algorithms*. MIT Press, 2nd edition edition, 2002.

[CLRS09] Thomas Cormen, Charles Leiserson, Ronald Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, third edition, 2009.

[CM14] Merlin Carl and Boris Zelikovich Moroz. On a diophantine representation of the predicate of provability. *Journal of Mathematical Sciences*, 199(1):36–52, 2014.

[CO08] Frédéric Chazal and Steve Yann Oudot. Towards persistence-based reconstruction in euclidean spaces. In *Proceedings of the twenty-fourth annual symposium on Computational geometry*, pages 232–241. ACM, 2008.

[CSEH07] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of Persistence Diagrams. *Discrete & Computational Ceometry*, 37(1):103–120, 2007.

[CSEM06] David Cohen-Steiner, Herbert Edelsbrunner, and Dmitriy Morozov. Vines and Vineyards by Updating Persistence in Linear Time. In *22nd Annual ACM Symposium on Computational Geometry*, pages 119–126, 2006.

[DE95] Cecil Jose A Delfinado and Herbert Edelsbrunner. An incremental algorithm for betti numbers of simplicial complexes on the 3-sphere. *Computer Aided Geometric Design*, 12(7):771–784, 1995.

[dFR85] Hubert de Fraysseix and Pierre Rosenstiehl. A characterization of planar graphs by trémaux orders. *Combinatorica*, 5(2):127–135, 1985.

[DHSW03] Jean-Guillaume Dumas, Frank Heckenbach, David Saunders, and Volkmar Welker. Computing simplicial homology based on efficient smith normal form algorithms. In *Algebra, Geometry and Software Systems*, pages 177–206. Springer, 2003.

[Die05] Reinhard Diestel. *Graph theory*. Springer Verlag, 2005.

[DM68] P. H. Doyle and D. A. Moran. A short proof that compact 2-manifolds can be triangulated. *Inventiones Mathematicae*, 5(2):160–162, 1968.

[DPeK82] Narsingh Deo, Gurpur M. Prabhu, and Mukkai S. et Krishnamoorty. Algorithms for generating fundamental cycles in a graph. *ACM Trans. Math. Software*, 8:26–42, 1982.

[DS95] Tamal K. Dey and Haijo Schipper. A new technique to compute polygonal schema for 2-manifolds with application to null-homotopy detection. *Discrete and Computational Geometry*, 14:93–110, 1995.

[DSG07] Vin De Silva and Robert Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7(1):339–358, 2007.

[DWW00] Colm Ó Dúnlaing, Colum Watt, and David Wilkins. Homeomorphism of 2-complexes is equivalent to graph isomorphism. *International Journal of Computational Geometry & Applications*, 10(5):453–476, 2000.

[Edm60] Jack R. Edmonds. A combinatorial representation of polyhedral surfaces. *Notices of the American Society*, 7:646, 1960.

[ELZ00] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological Persistence and Simplification. In *IEEE Symposium on Foundations of Computer Science*, pages 454–463, 2000.

[Eri12] Jeff Erickson. Combinatorial optimization of cycles and bases. In Afra Zomorodian, editor, *Advances in Applied and Computational Topology.*, Proceedings of Symposia in Applied Mathematics, 2012.

[EW05] Jeff Erickson and Kim Whittelsey. Greedy optimal homotopy and homology generators. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1038–1046, 2005.

[EW13] Jeff Erickson and Kim Whittelsey. Transforming curves on surfaces redux. In *Proc. of the 24rd Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1646–1655, 2013.

[Fár48] István Fáry. On straight line representation of planar graphs. *Acta sci. math. (Szeged)*, 1:229–233, 1948.

[Far08] Michael Farber. *Invitation to topological robotics*, volume 8. European Mathematical Society, 2008.

[Fre82] Michael Hartley Freedman. The topology of four-dimensional manifolds. *J. Differential Geom.*, 17(3):357–453, 1982.

[GL89] Cameron Gordon and John Luecke. Knots are determined by their complements. *J. Amer. Math. Soc.*, 2(2):371–415, 1989.

[GS90] Steve M. Gersten and Hamish B. Short. Small cancellation theory and automatic groups. *Inventiones mathematicae*, 102:305–334, 1990.

[GW01] Igor Guskov and Zoë Wood. Topological Noise Removal. In B. Watson and J. W. Buchanan, editors, *Proceedings of Graphics Interface 2001*, pages 19–26, 2001.

[Hak61] Wolfgang Haken. Theorie der Normalflachen, ein Isotopiekriterium für den Kreisnoten. *Acta Mathematica*, 105:245–375, 1961.

[Hal07a] Thomas C. Hales. The Jordan Curve Theorem, Formally and Informally. *The American Mathematical Monthly*, 114:882–894, dec 2007.

[Hal07b] Thomas C. Hales. Jordan's Proof of the Jordan Curve Theorem. *STUDIES IN LOGIC, GRAMMAR AND RHETORIC*, 10(23):45–60, 2007.

[Hat02] Allen Hatcher. *Algebraic topology*. Cambridge University Press, 2002. Available at http://www.math.cornell.edu/~hatcher/.

[Hef91] Lothar Heffter. Über das Problem der Nachbargebiete. *Math. Ann*, 38:477–508, 1891.

[Hef98] Lothar Heffter. Über metacyklische Gruppen und Nachbarconfigurationen. *Math. Ann*, 50:261–268, 1898.

[HLP99] Joel Hass, Jeffrey C. Lagarias, and Nicholas Pippenger. The computational complexity of knot and link problems. *Journal of the ACM*, 46(2):185–211, 1999.

[HM93] David Hartvigsen and Russel Mardon. When Do Short Cycles Generate the Cycle Space. *Journal of Combinatorial Theory, Series B*, 57:88–99, 1993.

[Hor87] Joseph D. Horton. A Polynomial-Time Algorithm to Find the Shortest Cycle Basis of a Graph. *SIAM Journal of Computing*, 16(2):358–366, 1987.

[HT74] J. Hopcroft and R. Tarjan. Efficient planarity testing. *Journal of the ACM*, 21(4):549–568, 1974.

[Jac05] William Jaco. Peking summer school, 2005. Lecture Notes available at https://www.math.oakstate.edu/~jaco/pekinglectures.htm.

[Jam99] I. M. James, editor. *History of topology*. Elsevier, 1999.

[Jon82] James P Jones. Universal diophantine equation. *The journal of symbolic logic*, 47(03):549–571, 1982.

[JT95] William Jaco and Jeffrey L. Tollefson. Algorithms for the complete decomposition of a closed 3-manifold. *Illinois Journal of Mathematics*, 39(3):358–406, 1995.

[KLM+09] Telikepalli Kavitha, Christian Liebchen, Kurt Mehlhorn, Dimitrios Michail, Romeo Rizzi, Torsten Ueckerdt, and Katharina A Zweig. Cycle bases in graphs characterization, algorithms, complexity, and applications. *Computer Science Review*, 3(4):199–243, 2009.

[KMMP04] Telikepalli Kavitha, Kurt Mehlhorn, Dimitrios Michail, and Katarzyna Paluch. A Faster Algorithm for Minimum Cycle Basis of Graphs. In *ICALP*, 2004.

[KMMP08] Telikepalli Kavitha, Kurt Mehlhorn, Dimitrios Michail, and Katarzyna E. Paluch. An Õ(m2 n) algorithm for minimum cycle basis of graphs. *Algorithmica*, 52:333–349, 2008.

[KMP77] Donald E. Knuth, James H. Morris, Jr, and Vaughan R. Pratt. Fast pattern matching in strings. *SIAM journal on computing*, 6(2):323–350, 1977.

[KMR08] Ken-ichi Kawarabayashi, Bojan Mohar, and Bruce Reed. A simpler linear time algorithm for embedding graphs into an arbitrary surface and the genus of graphs of bounded tree-width. In *49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 771–780. IEEE, 2008.

[Kup14] Greg Kuperberg. Knottedness is in NP, modulo GRH. *Advances in Mathematics*, 256:493–506, 2014.

[Kup15a] Greg Kuperberg. Algorithmic homeomorphism of 3-manifolds as a corollary of geometrization. *arXiv:1508.06720*, 2015.

[Kup15b] Greg Kuperberg. Algorithmic homeomorphism of 3-manifolds as a corollary of geometrization. Technical report, arXiV, 2015. https://arxiv.org/abs/1508.06720.

[Lac15] Marc Lackenby. A polynomial upper bound on Reidemeister moves. *Ann. Math. (2)*, 182(2):491–564, 2015.

[Lac16] Marc Lackenby. The efficient certification of knottedness and thurston norm. arXiv:1604.00290, 2016.

[Laz12] Francis Lazarus. Géométrie algorithmique. Notes de cours. https://pagesperso.g-scop.grenoble-inp.fr/~lazarusf/Enseignement/geoAlgo.html, 2012.

[Laz14] Francis Lazarus. *Combinatorial Graphs and Surfaces from the Computational and Topological Viewpoint Followed by some notes on The Isometric Embedding of the square Flat Torus*. Mémoire d'habilitation à diriger des recherches, Université Joseph Fourier, September 2014.

[Lei84] Frank Thomson Leighton. New lower bound techniques for vlsi. *Mathematical Systems Theory*, 17(1):47–70, 1984.

[LGQ09] Xin Li, Xianfeng Gu, and Hong Qin. Surface mapping using consistent pants decomposition. *IEEE Transactions on Visualization and Computer Graphics*, 15(4):558–571, 2009.

[Lov05] László Lovász. Graph minor theory. *Bulletin of the AMS*, 43(1):75–86, 2005.

[LR12] Francis Lazarus and Julien Rivaud. On the homotopy test on surfaces. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 440–449, 2012.

[Man14] Ciprian Manolescu. Triangulations of manifolds. *ICCM Not*, 2(2):21–23, 2014.

[MM09] Kurt Mehlhorn and Dimitrios Michail. Minimum Cycle Bases: Faster and Simpler. *ACM Transactions on Algorithms*, 6(1):art. 8, 2009.

[Moh99] Bojan Mohar. A linear time algorithm for embedding graphs in an arbitrary surface. *SIAM J. Discrete Math.*, 12:6–26, 1999.

[Moi52] Edwin E. Moise. Affine structures in 3-manifolds. V. The triangulation theorem and Hauptvermutung. *Ann. of Math. (2)*, 56:96–114, 1952.

[Moi77] Edwin E. Moise. *Geometric topology in dimensions 2 and 3*. Springer-Verlag, 1977.

[MT01] Bojan Mohar and Carsten Thomassen. *Graphs on Surfaces*. John Hopkins University Press, 2001.

[MVV87] Ketan Mulmuley, Umesh V. Vazirani, and Vijay V. Vazirani. Matching is as easy as matrix inversion. *Combinatorica*, 7(1):105–113, 1987.

[Pat13] Maurizio Patrignani. *Handbook of graph drawing and visualization, Roberto Tamassia editor*, chapter Planarity testing and embedding, pages 1–42. CRC press, 2013. https://cs.brown.edu/ rt/gdhandbook/.

[Per02] Grisha Perelman. The entropy formula for the Ricci flow and its geometric application. arXiv:math/0211159, 2002.

[Per03] Grisha Perelman. Finite extinction time for the solutions to the Ricci flow on certain three-manifolds. arXiv:math/0307245, 2003.

[Poo14] Bjorn Poonen. *Interpreting Gödel: Critical Essays*, chapter Undecidable problems: a sampler, pages 211–241. Cambridge University Press, 2014.

[RCS+97] AA Ranicki, A Casson, D Sullivan, M Armstrong, C Rourke, and G Cooke. The hauptvermutung book. *Collection of papers by Casson, Sullivan, Armstrong, Cooke, Rourke and Ranicki, K-Monographs in Mathematics*, 1, 1997.

[Rei27] Kurt Reidemeister. Elementare begründung der knotentheorie. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 5(1):24–32, 1927.

[RG96] Jürgen Richter-Gebert. *Realization spaces of Polytopes*, volume 1643 of *Lecture Notes in Mathematics*. Springer, Berlin, 1996.

[Rin74] Gerhard Ringel. *Map Color Theorem*. Springer-Verlag, 1974.

[Rob99] Vanessa Robins. Towards computing homology from finite approximations. *Topology proceedings*, 24, 1999.

[Sti82] John Stillwell. The word problem and the isomorphism problem for groups. *Bull. Amer. Math. Soc.*, 6(1):33–56, 1982.

[Sti87] John Stillwell. *Papers on group theory and topology*. Springer-Verlag, New York, 1987.

[Sti93] John Stillwell. *Classical topology and combinatorial group theory*. Springer-Verlag, New York, second edition, 1993.

[STP94] John Shawe-Taylor and Tomaž Pisanski. Homeomorphism of 2-complexes is graph isomorphism complete. *SIAM J. Comput.*, 23(1):120–132, 1994.

[Tar83] Robert E. Tarjan. *Data Structures and Network Algorithms*. Number 44 in CBMS-NFS Regional conference series in applied mathematics. SIAM, 1983.

[Tho92] Carsten Thomassen. The Jordan-Schönflies Theorem and the Classification of Surfaces. *American Mathematical Monthly*, 99(2):116–130, feb 1992.

[Tho93] Carsten Thomassen. Triangulating a surface with a prescribed graph. *Journal of Combinatorial Theory, Series B*, 57(2):196–206, 1993.

[Tut63] William T. Tutte. How to Draw a Graph. *Proc. London Mathematical Society*, 13:743–768, 1963.

[Tve80] Helge Tverberg. A Proof of the Jordan Curve Theorem. *Bull. London Math. Soc.*, 12:34–38, 1980.

[VZGG13] Joachim Von Zur Gathen and Jürgen Gerhard. *Modern computer algebra*. Cambridge university press, 2013.

[Wes01] Douglas B. West. *Introduction to Graph Theory*. Prentice Hall, second edition, 2001.

[Whi58] E. F. Whittelsey. Classification of finite 2-complexes. *Proceedings of the American Mathematical Society*, 9(6):841–845, 1958.

[Whi60] E. F. Whittlesey. Finite surfaces. a study of finite 2-complexes. part ii. the canonical form. *Mathematics Magazine*, 34(2):67–80, 1960.

[Wil96] Robin J. Wilson. *Introduction to Graph Theory*. Prentice Hall, fourth edition, 1996.

[Zie95] Günter M. Ziegler. *Lectures on polytopes*. Number 152 in Graduate texts in mathematics. Springer, rev. first ed edition, 1995.