Automorphisms and Isomorphisms of Maps in Linear Time

Ken-ichi Kawarabayashi ✉
National Institute of Informatics, Tokyo, Japan

Bojan Mohar ✉
Department of Mathematics, Simon Fraser University, Burnaby, Canada
IMFM, Department of Mathematics, University of Ljubljana, Slovenia

Roman Nedela ✉
Univeristy of West Bohemia, Pilsen, Czech Republic

Peter Zeman ✉
Department of Applied Mathematics, Faculty of Mathematics and Physics,
Charles University, Prague, Czech Republic

Abstract

A map is a 2-cell decomposition of a closed compact surface, i.e., an embedding of a graph such that every face is homeomorphic to an open disc. An automorphism of a map can be thought of as a permutation of the vertices which preserves the vertex-edge-face incidences in the embedding. When the underlying surface is orientable, every automorphism of a map determines an angle-preserving homeomorphism of the surface. While it is conjectured that there is no "truly subquadratic" algorithm for testing map isomorphism for unconstrained genus, we present a linear-time algorithm for computing the generators of the automorphism group of a map, parametrized by the genus of the underlying surface. The algorithm applies a sequence of local reductions and produces a uniform map, while preserving the automorphism group. The automorphism group of the original map can be reconstructed from the automorphism group of the uniform map in linear time. We also extend the algorithm to non-orientable surfaces by making use of the antipodal double-cover.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases maps on surfaces, automorphisms, isomorphisms, algorithm

Digital Object Identifier 10.4230/LIPIcs.ICALP.2021.86

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2008.01616`

Funding *Ken-ichi Kawarabayashi*: JSPS Kakenhi Grant Number JP18H05291 and 20H05965
Bojan Mohar: Supported in part by the NSERC Discovery Grant R611450 (Canada), and by the Research Project J1-2452 of ARRS (Slovenia).
Roman Nedela: Supported by Slovak Research and Development Agency under Grant No. APVV-19-0308 and by GAČR 20-15576S.
Peter Zeman: Supported by GAUK 1224120 and GAČR 20-15576S.

1 Introduction

The graph isomorphism problem asks whether or not two given graphs are isomorphic. It is one of the most fundamental problems in the theory of algorithms. It is probably the most notorious problem whose computational complexity is still a huge open question, even after Babai's recent quasipolynomial-time breakthrough [2]. While some complexity theoretic results indicate that this problem is unlikely NP-complete (if it was, the polynomial hierarchy would collapse to its second level, see [28]), no polynomial-time algorithm is known, even with extended resources like randomization or quantum computing.

On the other hand, there is a number of important classes of graphs on which the graph isomorphism problem is known to be solvable in polynomial time. These include graphs with bounded degree [23, 9], bounded eigenvalue multiplicity [3], bounded tree-width [22, 10], excluded small minors [11], etc.

In this paper, we are interested in planar graphs and, more generally, graphs of bounded genus. In 1966, Weinberg [30] gave a very simple quadratic algorithm for the graph isomorphism of planar graphs. This was improved by Hopcroft and Tarjan [16] to $\mathcal{O}(n \log n)$. Building, on this earlier work, Hopcroft and Wong [17] published in 1974 a paper, where they described a linear-time algorithm for isomorphism testing of planar graphs.

For graphs on surfaces of higher genus, the graph isomorphism problem seems much harder. This can be perhaps explained in the following way. We can rather easily reduce the problem to 3-connected graphs. For planar graphs, the famous result of Whitney [31] says that embeddings of 3-connected planar graphs in the plane are (combinatorially) unique. But for every simply connected surface, there exist 3-connected graphs with exponentially many embeddings. This makes an essential difference between planar graphs and graphs of higher genus.

For quite a long time it has been known that the isomorphism of bounded genus graphs can be solved in time $n^{\mathcal{O}(g)}$, where g is the genus of the underlying surface; see for example [27]. However, an interesting question is whether the result of Hopcroft and Wong [17] can be generalized also for the bounded genus graphs, i.e., whether the isomorphism problem for graphs of bounded genus can be solved in time $f(g) \cdot n$, for some computable function f. This motivates the study of the isomorphism problem for embedded graphs first.

By a *topological map* we mean a 2-cell decomposition of a closed compact surface, i.e., an embedding of a graph into a surface such that every face is homeomorphic to an open disc. An *isomorphism* of two maps is an isomorphism of the underlying graphs, which preserves the vertex-edge-face incidences. In particular, a map isomorphism induces a homeomorphism of the underlying surfaces. Our main result reads as follows.

▶ **Theorem 1.** *Let M_1 and M_2 be maps on a surface of genus g. The set of all isomorphisms* $\mathrm{Iso}(M_1, M_2)$ *from M_1 to M_2 can be determined in time $f(g) \cdot (\|M_1\| + \|M_2\|)$, where f is some computable function and $\|M\|$ denotes the size of the map M.*

In [21], two of the authors deal with a much weaker version of this problem, where only testing isomorphism is considered instead of constructing the whole set $\mathrm{Iso}(M_1, M_2)$. Recently, a linear-time algorithm was announced [19] for testing isomorphism of bounded genus graphs and the proposed approach heavily relies on our result. It should be also mentioned, that an algorithm with running time $n^{\mathcal{O}((\log g)^c)}$ for bounded genus graphs follows from [26], however, this result is based on completely different techniques.

Determining the set of all isomorphisms between two maps is closely related to finding the generators of the automorphism group $\mathrm{Aut}(M)$ of a map M, where an automorphism of M is just an isomorphism $M \to M$. More precisely, the set of all isomorphisms $M_1 \to M_2$ can be expressed as a composition $\psi \cdot \mathrm{Aut}(M_1)$ where $\psi : M_1 \to M_2$ is any isomorphism. Thus, our first result goes hand-in-hand with the following.

▶ **Theorem 2.** *Let M be a map on a surface of genus g. The generators of the automorphism group* $\mathrm{Aut}(M)$ *of M can be computed in time $f(g) \cdot \|M\|$, where f is some computable function and $\|M\|$ denotes the size of the map M.*

Colbourn and Booth [7] proposed a way to modify the Hopcroft-Wong algorithm [17] to compute the generators of the automorphism group of a spherical map. However, they state the following: "*We ... base our automorphism algorithms on the Hopcroft-Wong algorithm. Necessarily, we will only be able to sketch our procedure. A more complete description and a* *proof of correctness would require a more thorough analysis of the Hopcroft-Wong algorithm than has yet appeared in the literature.*" Sadly, the situation has not changed since, and the only available description of the Hopcroft-Wong algorithm is the extended abstract [17], which contains no proof of correctness and running time.[1] Our contribution also fills in this gap and we obtain much better insight into the Hopcroft-Wong algorithm by solving the problem in a much greater generality; see [20] as well.

Roughly speaking, the key idea of the Hopcroft-Wong algorithm is to try to apply contractions of edges to obtain two smaller isomorphic maps. In order to do this, edges must be chosen canonically, which is not always possible. Since Hopcroft and Wong consider only the spherical case, this situation occurs only in one special case. However, on the surfaces of higher genus, this situation is quite common and requires a completely different, more systematic, approach. As a consquence of considering the problem on the higher genus, our approach turnes out to be much simpler even for planar graphs than the approach originally proposed by Colbourn and Booth [7].

The Hopcroft-Wong algorithm reduces spherical maps to maps having the same degrees of vertices and also the same degrees of faces (e.g. Platonic solids). These maps are then treated separately. We, however, relax this condition and instead reduce our map to a map having the same cyclic vector of face sizes at each vertex (e.g. on sphere these also include Archimedean solids). The number of such maps is bounded for surfaces of genus $g > 1$, and for surfaces of genus $g \leq 1$ we give some special algorithms. This, surprisingly, allows a much more unified method of reducing the map, while preserving its automorphisms and isomorphisms.

Simultaneous conjugation problem. The problems of testing isomorphism of maps and computing the generators of the automorphism group of a map are related to the problem of *simultaneous conjugation*. In the latter problem, the input consists of two sets of permutations $\alpha_1, \dots, \alpha_d$ and $\beta_1, \dots, \beta_d$ on the set $\{1, \dots, n\}$, each of which generates a transitive subgroup of the symmetric group. The goal is to find a permutation γ such that $\gamma\alpha_i\gamma^{-1} = \beta_i$, for $i = 1, \dots, d$. Let us observe that this problem is a generalization of the map isomorphism problem. If α_1 and β_1 are involutions, $d = 2$, and the set $\{1, \dots, n\}$ is identified with the set of darts of a map on a surface (see Section 2 for definitions), then this problem is exactly the map isomorphism problem. If further $\alpha_1 = \beta_1$ and $\alpha_2 = \beta_2$, we get the map automorphism problem.

Since mid 1970s it has been known that the simultaneous conjugation problem can be solved in time $\mathcal{O}(dn^2)$ [8, 15]. A faster algorithm, with running time $\mathcal{O}(n^2 \log d / \log n + dn \log n)$, was found only recently [6]. This implies an $\mathcal{O}(n^2/\log n)$ algorithm for the isomorphism and automorphism problems for maps of unrestricted genus. In complexity theory, this is not considered to be a "truly subquadratic" algorithm. This motivates the following conjecture.

▶ **Conjecture 3.** *There is no $\varepsilon > 0$ for which there is an algorithm for testing isomorphism of maps of unrestricted genus in time $\mathcal{O}(n^{2-\varepsilon})$.*

An interesting open subproblem is to prove a conditional "truly superlinear" lower bound for any of the mentioned problems. There has been some progress in the direction of providing a lower bound. In particular it is known that the communication complexity of the simultaneous conjugation problem is $\Omega(dn \log(n))$, for $d > 1$, and that under the decision tree model the search version of the simultaneous conjugation problem has lower bound of $\Omega(n \log n)$ [5].

[1] The PhD thesis of Wong also does not bring any new information compared to [17].

2 Preliminaries

A *map* M is an embedding $\iota\colon X \to S$ of a connected graph X to a closed connected compact surface S such that every connected component of $S \setminus \iota(X)$ is homeomorphic to an open disc. The connected components are called *faces*. By $V(M)$, $E(M)$, and $F(M)$ we denote the sets of vertices, edges, and faces of M, respectively. We put $v(M) := |V(M)|$, $e(M) := |E(M)|$, and $f(M) := |F(M)|$.

Recall that closed connected compact surfaces are characterized by two invariants: orientability and the Euler characteristic χ. For the orientable surfaces, the latter can be replaced by the *(orientable) genus* $g \geq 0$, which is the number of tori in the connected sum decomposition of the surface, and for the non-orientable surfaces by the *non-orientable genus* $\gamma \geq 1$, which is the number of real projective planes in the connected sum decomposition of the surface. The following is well-known.

▶ **Theorem 4** (Euler-Poincaré formula). *Let M be a map on a surface S. Then $v(M) - e(M) + f(M) = \chi(S) = 2 - 2g$ if S has genus g and $\chi(S) = 2 - \gamma$ if S has non-orientable genus γ.*

We give an algebraic description of a map, where a map is defined by means of three fixed-point-free involutions acting on *flags*. A flag is a triple representing a vertex-edge-face incidence. The involutions are simply instructions on how to join the flags together to form a map. There are several advantages: (i) in such a form, maps can be easily passed to an algorithm as an input, (ii) verifying whether a mapping is an automorphism reduces to checking three commuting rules, and (iii) group theory techniques can be applied to obtain results about maps. For more details see for example [18] and [13, Section 7.6].

Oriented maps. Even though our main concern is in general maps, a large part of our algorithm deals with maps on orientable surfaces, where the algebraic description is simpler. An *oriented map* is a map on an orientable surface with a fixed global orientation. Every oriented map can be combinatorially described as a triple (D, R, L). Here, D is the set of *darts*. By a dart we mean an edge endowed with one of two possible orientations. Hence, each edge gives rise to two darts. The permutation $R \in \mathrm{Sym}(D)$, called *rotation*, is the product $R = \Pi_{v \in V} R_v$, where each R_v cyclically permutes the darts originating at $v \in V$, following the chosen orientation around v. The *dart-reversing involution* $L \in \mathrm{Sym}(D)$ is an involution of D that, for each edge, swaps the two oppositely oriented darts arising from the edge.

Formally, a *combinatorial oriented map* is any triple $M = (D, R, L)$, where D is a finite non-empty set of *darts*, R is any permutation of darts, L is a fixed-point-free involution of D, and the group $\langle R, L \rangle \leq \mathrm{Sym}(D)$ is transitive on D. By the size $\|M\|$ of the map, we mean the number of darts $|D|$. We require transitivity because the maps are connected by definition.

The group $\langle R, L \rangle$ is called the *monodromy group* of M. The vertices, edges, and faces of M are in one-to-one correspondence with the cycles of the permutations R, L, $R^{-1}L$, respectively. By the phrase "a dart x is incident to a vertex v" we mean that $x \in R_v$. Similarly, "x is incident to a face f" means that x belongs to the boundary walk of f defined by the respective cycle of $R^{-1}L$. By the *degree of a face* we mean the length of its boundary walk. A face of degree d will be called a d-face. Note that each dart is incident to exactly one face. For convenience, we frequently use a shorthand notation $x^{-1} = Lx$, for $x \in D$. The *dual* of an oriented map $M = (D, R, L)$ is the oriented map $M^* = (D, R^{-1}L, L)$.

Apart from standard map theory references, we need to introduce labeled maps. A *planted tree* is a rooted tree embedded in the sphere, i.e., a planted tree is a spherical map having exactly one face. We say that a planted tree is *integer-valued* if an integer is assigned to each vertex. A *dart-labeling* of an oriented map $M = (D, R, L)$ is a mapping $\ell\colon D \to \mathcal{T}$, where $\mathcal{T}$ is the set of integer-valued planted trees. A *labeled oriented map* M is a 4-tuple (D, R, L, ℓ). The *dual map* is the map M^* defined as $M^* = (D, R^{-1}L, L, \ell)$.

Two labeled oriented maps $M_1 = (D_1, R_1, L_1, \ell_1)$ and $M_2 = (D_2, R_2, L_2, \ell_2)$ are *isomorphic*, in symbols $M_1 \cong M_2$, if there exists a bijection $\psi\colon D_1 \to D_2$, called an *orientation-preserving isomorphism* from M_1 to M_2, such that

$$\psi R_1 = R_2\psi, \quad \psi L_1 = L_2\psi, \quad \text{and} \quad \ell_1 = \ell_2\psi. \tag{1}$$

The *set of orientation-preserving isomorphisms* from M_1 to M_2 is denoted by $\mathrm{Iso}^+(M_1, M_2)$. The *orientation-preserving automorphism group* of M is the set $\mathrm{Aut}^+(M) := \mathrm{Iso}^+(M, M)$. The following statement, which can be easily seen for unlabeled maps, extends also to labeled maps.

▶ **Theorem 5.** *Let M_1 and M_2 be labeled oriented maps with sets of darts D_1 and D_2, respectively. For every $x \in D_1$ and every $y \in D_2$, there exists at most one isomorphism $M_1 \to M_2$ mapping x to y. In particular, $\mathrm{Aut}^+(M_1)$ is fixed-point-free on D_1.*

▶ **Corollary 6.** *Let M_1 and M_2 be labeled oriented maps with sets of darts D_1 and D_2, respectively. If $x \in D_1$ and $y \in D_2$, then it can be checked in time $\mathcal{O}(|D_1| + |D_2|)$ whether there is an isomorphism mapping x to y.*

Chirality. The *mirror image* of an oriented map $M = (D, R, L)$ is the oriented map $M^{-1} = (D, R^{-1}, L)$. Similarly, the *mirror image* of labeled oriented map $M = (D, R, L, \ell)$ is the map $M^{-1} = (D, R^{-1}, L, \ell^{-1})$, where $\ell^{-1}(x)$ is the mirror image of $\ell(x)$ for each $x \in D$.

An oriented map M is called *reflexible* if $M \cong M^{-1}$. Otherwise the maps M and M^{-1} form a *chiral pair*. For example, all the Platonic solids are reflexible. The set of *all isomorphisms* from M_1 to M_2 is defined as $\mathrm{Iso}(M_1, M_2) := \mathrm{Iso}^+(M_1, M_2) \cup \mathrm{Iso}^+(M_1, M_2^{-1})$. Similarly, we put $\mathrm{Aut}(M) := \mathrm{Iso}(M, M)$.

Maps on all surfaces. Let M be a map on any, possibly non-orientable, surface. In general, a *combinatorial non-oriented map* is a quadruple (F, λ, ρ, τ), where F is a finite non-empty set of *flags*, and $\lambda, \rho, \tau \in \mathrm{Sym}(F)$ are fixed-point-free[2] involutions such that $\lambda\tau = \tau\lambda$ and the group $\langle \lambda, \rho, \tau \rangle$ acts transitively on F. By the *size* $\|M\|$ of the map M we mean the number of flags $|F|$.

Each flag corresponds uniquely to a vertex-edge-face incidence triple (v, e, f). Geometrically, it can be viewed as the triangle defined by v, the center of e, and the center of f. The group $\langle \lambda, \rho, \tau \rangle$ is called the *non-oriented monodromy group* of M. The vertices, edges, and faces of M correspond uniquely to the orbits of $\langle \rho, \tau \rangle$, $\langle \lambda, \tau \rangle$, and $\langle \rho, \lambda \rangle$, respectively. Similarly, an *isomorphism* of two non-oriented maps M_1 and M_2 is a bijection $\psi\colon F_1 \to F_2$ which commutes with λ, ρ, τ. The even-word subgroup $\langle \rho\tau, \tau\lambda \rangle$ has index at most two in the monodromy group of M. If it is exactly two, the map M is called *orientable*. For every oriented map (D, R, L) it is possible to construct the corresponding non-oriented map

[2] It is possible to extend the theory to maps on surfaces with boundaries by allowing fixed points of λ, ρ, τ.

(F, λ, ρ, τ). Conversely, from an orientable non-oriented map (F, λ, ρ, τ) it is possible to construct two oriented maps (D^+, R, L) and (D^-, R^{-1}, L), where D^+ and D^- are the two orbits of the even word subgroup, and $L = \tau\lambda$, $R = \rho\tau$.

Test of orientability. For a non-oriented map $M = (F, \lambda, \rho, \tau)$, it is possible to test in linear time if M is orientable [12, 24]. The *barycentric subdivision* B of M is constructed by placing a new vertex in the center of every edge and face, and then joining the centers of faces with the incident vertices and with the center of the incident edges. The dual of B is a 3-valent map, i.e., every vertex is of degree 3.

▶ **Theorem 7.** *A map $M = (F, \lambda, \rho, \tau)$ is orientable if and only if the underlying 3-valent graph of the dual of the barycentric subdivision of M is bipartite.*

Light vertices. A map is called *face-normal*, if all its faces are of degree at least three. It is well-known that every face-normal map on the sphere or on the projective plane has a vertex of degree at most 5. The next theorem generalizes this for other surfaces.

▶ **Theorem 8.** *Let S be a closed compact surface with Euler characteristic $\chi(S) \le 0$ and let M be a face-normal map on S. Then there is a vertex of valence at most $6(1 - \chi(S))$.*

Proof. A bound for maximum degree is achieved by a triangulation, thus we may assume that M is a triangulation. We have $f = 2e/3$. By plugging this in the Euler-Poincaré formula and using the Handshaking lemma, we obtain $3v - \overline{d}v/2 = 3\chi(S)$, where $\overline{d}$ is the average degree. By manipulating the equality, we get $\overline{d} - 6 = -6\chi(S)/v$. Since $\chi(S) \le 0$, the right hand side is maximized for $v = 1$. We conclude that $\overline{d} \le 6(1 - \chi(S))$. ◀

A vertex is called *light* if it is of minimum degree, otherwise it is called *heavy*.

Uniform and homogeneous maps. Given a map on an orientable surface, the cyclic vector of degrees of faces incident with a vertex v, induced by the chosen global orientation, is called the *local type* of v. A map is *uniform*[3] if the local types of all vertices are the same. A map is *homogeneous* of type $\{k, \ell\}$ if every vertex is of degree k and every face is of degree ℓ.

A *dipole* is a 2-vertex spherical map dual to a spherical cycle. *A bouquet* is a one-vertex map that is a dual of a *planted star* (a tree with at most one vertex of degree > 1).

▶ **Example 9.** The face-normal uniform spherical maps are: the 5 Platonic solids, the 13 Archimedean solids, pseudo-rhombicuboctahedron, prisms, antiprisms, and cycles of length at least 3. It easily follows from Euler's formula that the spherical homogeneous maps are the 5 Platonic solids, cycles, and dipoles.

3 Overview of the algorithm

We provide a high-level overview of the whole algorithm determining the automorphism group of a map. The input consists of a non-oriented map given by the quadruple $N = (F, \lambda, \rho, \tau)$.

First, using Theorem 7, we test whether N is orientable or not. If the map is orientable, then we know that the underlying surface is orientable and we fix a global orientation of the surface. We construct two oriented maps $M = (D, R, L)$ and $M^{-1} = (D, R^{-1}, L)$ representing N.

[3] In [1] Babai uses the term semiregular instead of uniform.

We start by determining $\mathrm{Aut}^+(M)$. On the map M, we perform a sequence of elementary local reductions (Section 4). There are two types of reductions: normalization and elimination of vertices of minimum degree. The normalization is of the highest priority and its purpose is to ensure that the resulting map is face-normal. In a face-normal map, it is guaranteed by Theorem 8 that there is a vertex of small degree. The second elementary reduction replaces a vertex of minimum degree by a polygon connecting its higher-degree neighbours and reconnecting the other incident edges (see Figure 3). These two reductions are applied until we are left with a map which has all vertices of degree k. Now, we observe that our reductions do not really depend on the degrees of vertices, but rather on some vertex-labelling (not related to dart labelling) which is linearly ordered. At this stage we can no longer distinguish vertices based on their degreee. We refine the procedure by using the local types instead of degrees. Note that the local types can be linearly ordered. It follows from Theorem 8 that the number of local types sufficient to consider is bounded. Thus, our reduction can be applied in the same way, but instead of degrees we use local types. The result is a labeled face-normal uniform oriented map $M' = (D', R', L', \ell')$ with $\mathrm{Aut}^+(M) \cong \mathrm{Aut}^+(M')$ and $D' \subseteq D$; for more details see Section 4.

The number of face-normal uniform oriented map M' on a surface of genus $g > 1$ is bounded by a fucntion of g (Proposition 13), which means that a brute-force approach is sufficient to determine $\mathrm{Aut}^+(M')$. For the case of sphere and torus, the problem is non-trivial since there are infinite families of face-normal uniform maps and a special treatment is necessary; for more details see Section 5. Now, since $\mathrm{Aut}^+(M)$ acts fixed-point-freely on D and $D' \subseteq D$, there is a unique way to extend $\mathrm{Aut}^+(M')$ to $\mathrm{Aut}^+(M)$. Finally, to construct $\mathrm{Aut}(M)$, we run the whole algorithm again to determine $\mathrm{Iso}(M, M^{-1})$.

If the map is N is non-orientable, we construct its oriented antipodal double-cover $\widetilde{M} = (D, R, L) = (F, \rho\tau, \tau\lambda)$. We show that $\mathrm{Aut}(N) \le \mathrm{Aut}^+(\widetilde{M})$, and therefore, we can again apply our algorithm to determine $\mathrm{Aut}^+(\widetilde{M})$. Here, the most difficult part is to determine $\mathrm{Aut}(N)$ within $\mathrm{Aut}^+(\widetilde{M})$. For the case of projective plane and Klein bottle the problem is highly non-trivial and a special treatment is again needed, while for the other cases, again, a brute force approach is sufficient; for more details see Section 6.

4 From oriented maps to uniform oriented maps

In this section, we describe a set of elementary reductions defined on labeled oriented maps, given by a quadruple (D, R, L, ℓ), in detail. The output of each elementary reduction is always a quadruple (D', R', L', ℓ'), satisfying $D' \subseteq D$, $v(M') + e(M') < v(M) + e(M)$, and $\mathrm{Aut}^+(M') \cong \mathrm{Aut}^+(M)$. If none of the reductions apply, the map is a uniform oriented map. The procedure defines a function which assigns to a given oriented map M a unique labeled oriented map U with $\mathrm{Aut}^+(M) \cong \mathrm{Aut}^+(U)$. Since the darts of U form a subset of the darts of M, by semiregularity, every generator of $\mathrm{Aut}^+(U)$ can be extended to a generator of $\mathrm{Aut}^+(M)$ in linear time. We deal with the uniform oriented maps in Section 5.

After every elementary reduction, to ensure that $\mathrm{Aut}^+(M') = \mathrm{Aut}^+(M)$, we need to define a new labeling ℓ'. To this end, in the whole section, we assume that we have an injective function $\mathbf{Label}\colon \mathbb{N} \times \bigcup_{k=1}^{\infty} \mathcal{T}^k \to \mathcal{T}$, where $\mathcal{T}$ is the set of all integer-valued planted trees. Moreover, we assume that the root of $\mathbf{Label}(t, T_1, \dots, T_k)$ contains the integer t, corresponding to the current step of the reduction procedure. After every elementary reduction, this integer is increased by one; see the full version for more details.

Even though we defined our reductions only based on the minimum degree, it can be easily seen that we are only using the fact that natural numbers are linearly ordered. Thus, our reduction really works with any vertex lables, which are linearly ordered. In particular, if we replace degrees with local types together with a natural lexicographic linear ordering, our reductions are well-defined. The consequence is thet every irreducible map with respect to these reductions is a face-normal uniform map.

Normalization. By Theorem 8, there is always a light vertex in a face-normal map. The purpose of the following reduction is to remove faces of degree one and two. This reduction is of the highest priority and it is applied until the map is one of the following: (i) face-normal, (ii) bouquet, (iii) dipole. In the cases (ii) and (iii), the whole reduction procedure stops with a uniform map. In the case (i), the reduction procedure continues with further reductions. We describe the reduction formally.

For technical reasons we split the reduction into two parts: deletion of loops, denoted by $\mathbf{Loops}(M)$, and replacement of a dipole by an edge, denoted by $\mathbf{Dipoles}(M)$.

Reduction Loops. If $M = (D, R, L, \ell)$ with $v(M) > 1$ contains loops, we remove them. Let $\mathcal{L}$ be the list of all maximal sequences of darts of the form $s = \{x_1, x_1^{-1}, \dots, x_k, x_k^{-1}\}$, where $Rx_i = x_i^{-1}$, for $i = 1, \dots, k$, $Rx_i^{-1} = x_{i+1}$ for $i = 1, \dots, k-1$, and $Rx_k^{-1} \neq x_1$. By definition, $R^{-1}Lx_i = x_i$, hence x_i bounds a 1-face, for $i = 1, \dots, k-1$; see Figure 1. Moreover, for each such sequence s, all the darts x_i are incident to the same vertex $v \in V(M)$. We say that the unique vertex v with $R_v = (x_0, x_1, x_1^{-1}, \dots, x_k, x_k^{-1}, x_{k+1}, \dots)$ is *incident* to s. We call the darts x_0 and x_{k+1} the *bounding darts* of the sequence s.

The new map $M' = (D'; R', L', \ell') =: \mathbf{Loops}(M)$ is defined as follows. First, we put $D' := D \setminus \bigcup_{s \in \mathcal{L}} s$, and $L' := L_{\restriction D'}$. Let $s = \{x_1, x_1^{-1}, \dots, x_k, x_k^{-1}\} \in \mathcal{L}$ with bounding darts x_0 and x_{k+1}. If v is incident to s, then we put $R'_v := (x_0, x_{k+1}, \dots)$, else we put $R'_v := R_v$. Moreover, we put $\ell'(x_0) := \mathbf{Label}(t, a_0, \dots, a_k)$ and $\ell'(x_{k+1}) := \mathbf{Label}(t, a_{k+1}, b_k, \dots, b_1)$, where t is the current step, $a_i = \ell(x_i)$, for $i = 0, \dots, k+1$, and $b_i = \ell(x_i^{-1})$, for $i = 1, \dots, k$. For every $x \in D'$ which is not a bounding dart in M, we put $\ell'(x) := \ell(x)$. We obtain a well-defined map M' with no faces of valence one; see Figure 1.

▶ **Lemma 10.** *Let $M_i = (D_i, R_i, L_i, \ell_i)$, $i = 1, 2$ where $D_1 \cap D_2 = \emptyset$, be labeled oriented maps. Let $M'_1 := \mathbf{Loops}(M_1)$ and $M'_2 := \mathbf{Loops}(M_2)$. Then $\mathrm{Iso}^+(M_1, M_2)_{\restriction D'_1} = \mathrm{Iso}^+(M'_1, M'_2)$. In particular, $\mathrm{Aut}^+(M_1)_{\restriction D'_1} = \mathrm{Aut}^+(M'_1)$.*

Reduction Dipoles. If $M = (D, R, L, \ell)$ with $v(M) > 2$ contains dipoles. Let $\mathcal{L}$ be the list of all maximal sequences $s = (x_1, \dots, x_k)$ of darts, $k > 1$, satisfying $Rx_i = x_{i+1}$, $(R^{-1}L)^2 x_i = x_i$, and either $Rx_k \neq x_1$ or $Rx_1^{-1} \neq x_k^{-1}$; see Figure 2. Let $s^{-1} := (x_k^{-1}, \dots, x_1^{-1}) \in \mathcal{L}$ be the *inverse* sequence. There are vertices u and v such that $R_u = (y_1, s, y_2, \dots)$ and

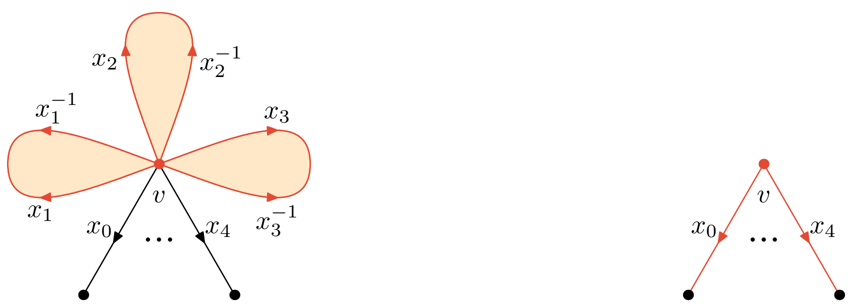

Figure 1 A sequence of darts $x_1, x_1^{-1}, x_2, x_2^{-1}, x_3, x_3^{-1}$ with bounding darts x_0 and x_4.

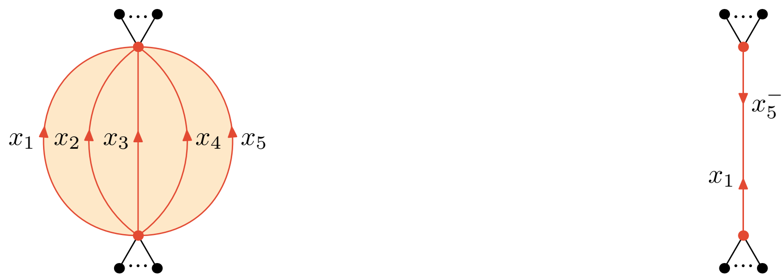

Figure 2 A sequence of darts $x_1, \dots, x_5$ forming a dipole.

$R_v = (z_1, s^{-1}, z_2, \dots)$, for some $y_1, y_2, z_1, z_2 \in D$. At least one of the sets $\{y_1, y_2\}$, $\{z_1, z_2\}$ is non-empty since otherwise $v(M) = 2$ and M is a dipole. We say that u, v are *incident* to s, s^{-1}, respectively; see Figure 2

The new map $M' = (D', R', L', \ell') =: \mathbf{Dipoles}(M)$ is defined as follows. First, we put

$$D' := D \setminus \bigcup_{(x_1,\dots,x_k)\in\mathcal{L}} \{x_2, \dots, x_k\} \cup \{x_1^{-1}, \dots, x_{k-1}^{-1}\}.$$

Let $s = (x_1, \dots, x_k) \in \mathcal{L}$. If u and v are incident to s and s^{-1}, respectively, then we put $R'_u := (y_1, x_1, y_2 \dots)$ and $R'_v := (z_1, x_k^{-1}, z_2, \dots)$, else we put $R'_u := R_u$. Next, we put $L'x_1 := x_k^{-1}$, $L'x_k^{-1} := x_1$, and $L'x := Lx$ if $x \notin s \in \mathcal{L}$. Finally, we put $\ell'(x_1) := \mathbf{Label}(t, a_1, \dots, a_k)$ and $\ell'(x_k^{-1}) := \mathbf{Label}(t, b_k, \dots, b_1)$, where t is the current step, $a_i = \ell(x_i)$ and $b_i = \ell(x_i^{-1})$, for $i = 1, \dots, k$. We put $\ell'(x) := \ell(x)$ for $x \notin s \in \mathcal{L}$. We obtain a well-defined map M' with no 2-faces; see Figure. 2.

▶ **Lemma 11.** *Let $M_i = (D_i, R_i, L_i, \ell_i)$, $i = 1, 2$ where $D_1 \cap D_2 = \emptyset$, be labeled oriented maps. Let $M'_1 := \mathbf{Dipoles}(M_1)$ and $M'_2 := \mathbf{Dipoles}(M_2)$. Then $\mathrm{Iso}^+(M_1, M_2)_{\restriction D'_1} = \mathrm{Iso}^+(M'_1, M'_2)$. In particular, $\mathrm{Aut}^+(M_1)_{\restriction D'_1} = \mathrm{Aut}^+(M'_1)$.*

Face-normal maps. The input is a labeled face-normal oriented map $M = (D, R, L, \ell)$ and a list $\mathcal{L}$ of all light vertices of degree d which have at least one heavy neighbour. For every vertex $v \in \mathcal{L}$, we denote by $u_0, \dots, u_{k-1}$, for some $1 \le k \le d$, the cyclic sequence of all heavy neighbours of v, following the prescribed orientation of the underlying surface. Denote by $x_0, x_1, \dots, x_{k-1}$ the darts based at $u_0, u_1, \dots, u_{k-1}$, joining u_j to v for $j = 0, \dots, k-1$. Let $R_{u_i} = (y_i, x_i, z_i, \dots)$, for $i = 0, \dots, k-1$, and let

$$R_v = (x_0^{-1}, A_0, x_1^{-1}, A_1, \dots, x_{k-1}^{-1}, A_{k-1}),$$

where each A_i is a (possibly empty) sequence of darts.

The new map $M' = (D', R', L', \ell') =: \mathbf{Delete}(M)$ is defined as follows. We set $D' := D$ and $L' := L$. For a heavy vertex w with no light neighbour, we have $R'_w := R_w$. If $v \in \mathcal{L}$, with the above notation, we set $R'_{u_i} := (y_i, A_i, x_i, x_{i-1}^{-1}, z_i, \dots)$. Moreover, we set $\ell'(x_i) := \mathbf{Label}(t, \ell(x_i))$ and $\ell'(x_i^{-1}) := \mathbf{Label}(t, \ell(x_i^{-1}))$, where t is the current step number; see Figure 3.

▶ **Lemma 12.** *Let $M_i = (D_i, R_i, L_i, \ell_i)$, $i = 1, 2$ where $D_1 \cap D_2 = \emptyset$, be labeled oriented maps. Let $M'_1 := \mathbf{Delete}(M_1)$ and $M'_2 := \mathbf{Delete}(M_2)$. Then $\mathrm{Iso}^+(M_1, M_2) = \mathrm{Iso}^+(M'_1, M'_2)$. In particular, $\mathrm{Aut}^+(M_1) = \mathrm{Aut}^+(M'_1)$.*

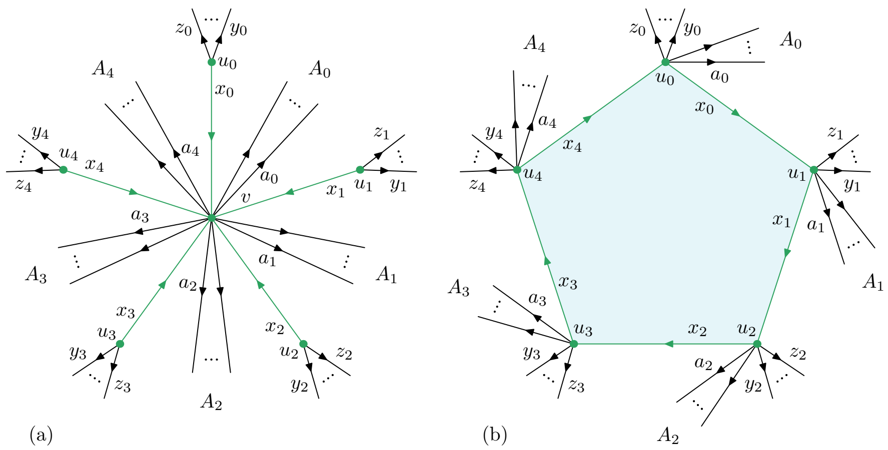

Figure 3 An example of the reduction deleting a vertex.

Proof. Let $\psi\colon M_1 \to M_2$ be an isomorphism. We prove that ψ is also an isomorphism of M_1' and M_2'. We check the commuting rules (1) for ψ. We have $L_i' = L_i$, for $i = 1, 2$, so $L_1'\psi = \psi L_2'$. For R_1' and R_2', we need to check the commuting rules only at $x_i, x_i^{-1}, y_i, a_i \in D_1'$, for $i = 0, \dots, k-1$, where a_i is the last dart in the sequence A_i. We have

$$\psi R_1' x_i = \psi R_1^{-1} L_1 x_i = R_2^{-1} L_2 \psi x_i = R_2' \psi x_i,$$

$$\psi R_1' x_i^{-1} = \psi R_1 L_1 x_i^{-1} = R_2 L_2 \psi x_i^{-1} = R_2' \psi x_i^{-1}.$$

It remains to check the commuting rules at each y_i and a_i. Note that if A_i is empty there is nothing to check. We have

$$\psi R_1' y_i = \psi R_1 L_1 R_1 y_i = R_2 L_2 R_2 \psi y_i = R_2' \psi y_i.$$

Further, using the relations $R_1' a_i = x_i = L_1 R_1^q a_i$, for some $q > 0$, we get

$$\psi R_1' a_i = \psi x_i = \psi L_1 R_1^q a_i = L_2 R_2^q \psi a_i = R_2' \psi a_i.$$

Putting it together, we proved that $\psi R_1' = R_2' \psi$. Clearly, $\ell_1'(x_i) = \ell_2'(\psi x_i)$ if and only if $\ell_1(x_i) = \ell_2(\psi x_i)$. Similarly for x_i^{-1}.

For the converse, we assume that $\psi R_1' = R_2' \psi$ and $\psi L_1' = L_2' \psi$ and we prove $\psi R_1 = R_2 \psi$ and $\psi L_1 = L_2 \psi$. Similarly as above, we need to check the commuting rules for $x_i, x_i^{-1}, y_i, a_i \in D_1$.

- By the definition of M_1' and M_2', we have $R_1 x_i = z_i = (R_1')^2 x_i$. Since **Label** is injective, we have $R_2 \psi x_i = \psi z_i = (R_2')^2 \psi x_i$. Using these relations, we get

$$\psi R_1 x_i = \psi (R_1')^2 x_i = (R_2')^2 \psi x_i = R_2 \psi x_i.$$

- By the definition of M_1' and M_2', we have $R_1 x_i^{-1} = R_1'^m L_1' x_i^{-1}$, for some m. Since **Label** is injective, we have $R_2 \psi x_i^{-1} = R_2'^m L_2' \psi x_i^{-1}$. Using these relations, we get

$$\psi R_1 x_i^{-1} = \psi R_1'^m L_1 x_i^{-1} = R_1'^m L_2' \psi x_i^{-1} = R_2 \psi x_i^{-1}.$$

- By the definition of M_1' and M_2', we have $R_1 y_i = x_i = R_1'^m y_i$, for some m. Since **Label** is injective, $R_2 \psi y_i = \psi x_i = R_2'^m \psi y_i$. Using these relations, we get

$$\psi R_1 y_i = \psi R_1'^m y_i = R_2'^m \psi y_i = R_2 \psi y_i.$$

- By the definition of M_1' and M_2', we have $R_1 a_i = L_1' R_1'^{-1} L_1' R_1' a_i$. Since **Label** is injective, $R_2 \psi a_i = L_2' R_2'^{-1} L_2' R_2' \psi a_i$. Using these relations, we get

$$\psi R_1 a_i = \psi L_1' R_1'^{-1} L_1' R_1' a_i = L_2' R_2'^{-1} L_2' R_2' \psi a_i = R_2 \psi a_i.$$

Putting it togehter, we proved that $\psi R_1 = R_2 \psi$, which implies that ψ is an isomorphism $M_1 \to M_2$. This completes the proof. ◀

5 Irreducible maps on orientable surfaces

In this section, we provide an algorithm computing the automorphism group of irreducible oriented maps, with fixed Euler characteristic, in linear time. The proof splits into three parts: maps of negative Euler characteristic, maps on the sphere, and maps on torus.

Surfaces of negative Euler characteristic. If the Euler characteristic χ is negative, the irreducible maps are exactly all the uniform face-normal maps. We prove that the number of uniform face-normal maps is bounded by a function of χ. Therefore, generators of the automorphism group can be computed by a brute force approach. Note that the following lemma does not require the underlying surface to be orientable, it only requires χ to be negative.

▶ **Proposition 13.** *The number of edges of a uniform face-normal map on a closed compact surface S with Euler characteristic $\chi(S) < 0$ is bounded by a function of $\chi(S)$.*

Proof. Babai noted in [1, Theorem 3.3] that the Hurwitz Theorem (see, e.g. [4] or [12]) implies that the number of vertices of a uniform map M on S is at most $84|\chi(S)|$. By Theorem 8, the degree of a vertex of M is bounded by a function of $\chi(S)$ as well. Therefore, the number of edges is also bounded by a function of $\chi(S)$ and the theorem follows. ◀

▶ **Corollary 14.** *Let $M = (D, R, L)$ be a uniform face-normal map $M = (D, R, L)$ on an orientable surface S with $\chi(S) < 0$. Then $\mathrm{Aut}(M)$ can be computed in time $f(\chi(S))|D|$, for some computable function f.*

Sphere. By the definition of the reductions in Section 4, the irreducible spherical maps are the five Platonic maps, 13 Archimedean maps, pseudo-rhombicuboctahedron, prisms, antiprisms, cycles, dipoles, and bouquets.

In the first three cases, the automorphism group can be computed by a brute force approach. We show that for (labeled) prisms, antiprisms, dipoles and bouquets, the problem can be reduced to computing the automorphism group of a vertex-labeled cycle.

▶ **Theorem 15** ([17])**.** *If $M = (D, R, L)$ is an irreducible spherical map, then the generators of $\mathrm{Aut}(M)$ can be computed in time $\mathcal{O}(|D|)$.*

Torus. The toroidal irreducible maps are uniform face-normal maps. The universal covers of uniform toroidal maps are uniform tilings (infinite maps with finite vertex and face degrees) of the Euclidean plane. There are 12 of such tilings; see [14, page 63]. The corresponding local types are $(3,3,3,3,3,3)$, $(4,4,4,4)$, $(6,6,6)$, $2 \times (3,3,3,3,6)$, $(3,3,3,4,4)$, $(3,3,4,3,4)$, $(3,4,6,4)$, $(3,6,3,6)$, $(3,12,12)$, $(4,6,12)$, and $(4,8,8)$. One type occurs in two forms, one of the respective tilings is the mirror image of the other. Each of these tilings T gives rise to an infinite family of toroidal uniform maps as follows. It is well-known that $\mathrm{Aut}^+(T)$ is isomorphic either to the triangle group $\Delta(4,4,2)$ or to $\Delta(6,3,2)$. Each of these contains an infinite subgroup H of translations generated by two shifts. Every finite uniform toroidal map of the prescribed local type can be constructed as the quotient T/K, where K is a subgroup of H of finite index.

First, our algorithm reduces a uniform map to one of the two homogeneous types $\{4,4\}$ and $\{6,3\}$, while preserving the automorphism group. Then, the algorithm computes the generators of the automorphism groups of a labeled homogeneous toroidal map M of type $\{4,4\}$ or $\{3,6\}$. For technical reasons, we transform the dart-labelling to a vertex-labelling of M. These transformations can be done easily by, for a given vertex, encoding the lables of the outgoing darts into the vertex. The following lemma describes some important properties of $\mathrm{Aut}^+(M)$.

▶ **Lemma 16** ([29]). *Let M be a toroidal map of type $\{4,4\}$ or $\{6,3\}$. The orientation-preserving automorphism group of a labeled map M is a semidirect product $T \rtimes H$, where T is a direct product of two cyclic groups, and $|H| \leq 6$. Moreover, the action of T is regular on the vertices of M.*

Since the order of H is bounded by a constant, it takes linear time to check whether every element of H is a label-preserving automorphism. The main difficulty is to find T. The subgroup T is generated by α and β, where α is the horizontal, and β is the vertical shift by the unit distance. Now the meaning of the parameters r, s, t is the following: $|\alpha| = r$, $\alpha^t = \beta^s$, and s is the least power of β such that $\beta^s \in \langle\alpha\rangle$. The following lemma shows that T can always be written as a direct product of two cyclic groups.

▶ **Lemma 17.** *There exists δ and γ such that $T = \langle\delta\rangle \times \langle\gamma\rangle$. Moreover, δ and γ can be computed in time $\mathcal{O}(rs)$.*

Lemma 17 can be viewed as a transformation of the shifted grid $\mathcal{G}$ to the orthogonal grid $\mathcal{G}^\perp$. Note that the underlying graph may change, but both $\mathcal{G}$ and $\mathcal{G}^\perp$ are Cayley graphs based on the group T, therefore, the vertex-labeling naturally transfers. Thus, we may assume that $t = 0$ and $T = \langle\alpha\rangle \times \langle\beta\rangle \cong \mathbb{Z}_r \times \mathbb{Z}_s$. We need to compute generators of the label-preserving subgroup of T.

From now on, we assume that we are given a *cyclic orthogonal grid* $\mathcal{G}$ of size rs, which is graph with vertices identified with $(i,j) \in G$, where $G = \mathbb{Z}_r \times \mathbb{Z}_s$. For every (i,j), there is an edge between (i,j) and $(i+1 \mod r, j)$, and between (i,j) and $(i, j+1 \mod s)$. Moreover, we are given an integer-labeling ℓ of the vertices of $\mathcal{G}$. Clearly, $\mathcal{G}$ determines the *ℓ-preserving subgroup* H of G, namely

$$H = \{(x,y) : \forall (i,j) \in G, \ell(i,j) = \ell(i+x, j+y)\}.$$

The goal is to find the generators of H in time $\mathcal{O}(rs)$.

We give a description of any subgroup of the direct product of G that is suitable for our algorithm. First, we define four important mappings. The two *projections* $\pi_1 \colon G \to \mathbb{Z}_r$ and $\pi_2 \colon G \to \mathbb{Z}_s$ are defined by $\pi_1(x,y) = x$ and $\pi_2(x,y) = y$, respectively. The two *inclusions* $\iota_1 \colon \mathbb{Z}_r \to G$ and $\iota_2 \colon \mathbb{Z}_s \to G$ are defined by $\iota_1(x) = (x,0)$ and $\iota_2(y) = (0,y)$, respectively.

▶ **Lemma 18.** *Let $G = \mathbb{Z}_r \times \mathbb{Z}_s$ for $r, s \geq 1$, and let H be a subgroup of G. Then there are $a, c \in \mathbb{Z}_r$ and $b \in \mathbb{Z}_s$ such that*

$$H = \{(ia + jc, jb) : i,j \in \mathbb{Z}\} = \langle (a,0), (c,b) \rangle,$$

where $\langle a\rangle = \iota_1^{-1}(H)$, $\langle b\rangle = \pi_2(H)$, and $c < a$ is the minimum integer such that $(c,b) \in H$.

This description suggests an algorithm to find the generators of the given subgroup H of $\mathbb{Z}_r \times \mathbb{Z}_s$. In our setting, the subgroup H is given on the input by a labeling function ℓ, defined on the vertices of the $r \times s$ orthogonal grid. The subgroup H is the ℓ-preserving subgroup of $\mathbb{Z}_r \times \mathbb{Z}_s$.

To compute the generators of H, it suffices, by Lemma 18, to determine $a, c \in \mathbb{Z}_r$ and $b \in \mathbb{Z}_s$ such that $\langle a \rangle = \iota_1^{-1}(H)$, $\langle b \rangle = \pi_2(H)$, and c is the smallest integer such that $(c, b) \in H$. Then $H = \langle (a, 0), (c, b) \rangle$.

▶ **Lemma 19.** *There is an $\mathcal{O}(rs)$-time algorithm which computes the integers a, b, c such that $\iota_1^{-1}(H) = \langle a \rangle$, $\pi_2(H) = \langle b \rangle$ and $c < a$ is the smallest integer such that $(c, b) \in H$.*

The results of this subsection are summarized by the following.

▶ **Theorem 20.** *If $M = (D, R, L, \ell)$ is a uniform face-normal labeled toroidal map, then the generators of $\mathrm{Aut}(M)$ can be computed in time $\mathcal{O}(|D|)$.*

6 Non-orientable surfaces

For a map M on a non-orientable surface S, we reduce the problem of computing the generators of $\mathrm{Aut}(M)$ to the problem of computing the generators of $\mathrm{Aut}^+(\widetilde{M})$, for some orientable map $\widetilde{M}$. In particular, the map $\widetilde{M}$ is the antipodal double cover of M.

Given a map $M = (F, \lambda, \rho, \tau)$ on a non-orientable surface of genus γ, we define the antipodal double cover $\widetilde{M} = (D, R, L)$ by setting $D := F$, $R := \rho\tau$, and $L := \tau\lambda$. Since M is non-orientable, we have $\langle R, L \rangle = \langle \lambda, \rho, \tau \rangle$, so $\langle R, L \rangle$ is transitive and $\widetilde{M}$ is well-defined. For more details on this construction see [25]. We note that $\widetilde{\chi} = 2\chi$, where $\widetilde{\chi}$ and χ is the Euler characteristic of the underlying surface of $\widetilde{M}$ and M, respectively.

▶ **Lemma 21.** *We have $\mathrm{Aut}(M) \le \mathrm{Aut}^+(\widetilde{M})$.*

Proof. Let $\varphi \in \mathrm{Aut}(M)$. Then we have $R^\varphi = (\rho\tau)^\varphi = \rho^\varphi\tau^\varphi = \rho\tau = R$ and $L^\varphi = (\tau\lambda)^\varphi = \tau^\varphi\lambda^\varphi = \tau\lambda = L$. ◀

▶ **Lemma 22.** *We have $\mathrm{Aut}(M) = \{\varphi \in \mathrm{Aut}^+(\widetilde{M}) : \varphi\tau = \tau\varphi\}$.*

Proof. Let $\varphi \in \mathrm{Aut}^+(\widetilde{M})$. We have $\varphi R \varphi^{-1} = R$ and $\varphi L \varphi^{-1} = L$. By plugging in $R = \rho\tau$ and $L = \tau\lambda$, we obtain

$$\varphi(\rho\tau)\varphi^{-1} = \rho\tau \quad \text{and} \quad \varphi(\tau\lambda)\varphi^{-1} = \lambda\tau.$$

From there, by rearranging the left-hand sides of the equations, we get

$$(\varphi\rho\varphi^{-1})(\varphi\tau\varphi^{-1}) = \varphi(\rho\tau)\varphi^{-1} = \rho\tau \quad \text{and} \quad (\varphi\tau\varphi^{-1})(\varphi\lambda\varphi^{-1}) = \varphi(\tau\lambda)\varphi^{-1} = \tau\lambda.$$

Finally, we obtain

$$\varphi\rho\varphi^{-1} = \rho\tau(\varphi\tau\varphi^{-1}) \quad \text{and} \quad \varphi\lambda\varphi^{-1} = (\varphi\tau\varphi^{-1})\tau\lambda.$$

If $\varphi \in \mathrm{Aut}(M)$, then, in particular, it commutes with τ. On the other hand, if φ commutes with τ, then the last two equations imply that it also must commute with ρ and λ, i.e., $\varphi \in \mathrm{Aut}(M)$. ◀

The previous lemmas are key and suggest an approach for computing the generators of the automorphism group of M. In particular, it is necessary to check which automorphisms of $\widetilde{M}$ commute with τ. The cases when the underlying surface is the projective plane, or the Klein bottle, must be treated separately.

▶ **Theorem 23.** *Let $M = (F, \lambda, \rho, \tau)$ be a map on a non-orientable a non-orietable surface of genus γ. Then it is possible to compute the generators of $\mathrm{Aut}(M)$ in time $f(\gamma)|F|$.*

References

1 L. Babai. Vertex-transitive graphs and vertex-transitive maps. *Journal of graph theory*, 15(6):587–627, 1991.

2 L. Babai. Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 684–697. ACM, 2016.

3 L. Babai, D. Y. Grigoryev, and D. M. Mount. Isomorphism of graphs with bounded eigenvalue multiplicity. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 310–324. ACM, 1982.

4 N. L. Biggs and A. T. White. *Permutation groups and combinatorial structures*, volume 33. Cambridge University Press, 1979.

5 A. Brodnik, A. Malnič, and R. Požar. Lower bounds on the simultaneous conjugacy problem in the symmetric group. In *4th annual Mississippi Discrete Mathematics Workshop*, 2015.

6 A. Brodnik, A. Malnič, and R. Požar. Fast permutation-word multiplication and the simultaneous conjugacy problem. *arXiv preprint*, 2019. `arXiv:1907.07889`.

7 C. J. Colbourn and K. S. Booth. Linear time automorphism algorithms for trees, interval graphs, and planar graphs. *SIAM Journal on Computing*, 10(1):203–225, 1981.

8 M. Fontet. Calcul de centralisateur d'un groupe de permutations. *Bull. Soc. Math. France Mém.,(49-50)*, pages 53–63, 1977.

9 M. Grohe, D. Neuen, and P. Schweitzer. A faster isomorphism test for graphs of small degree. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 89–100, 2018.

10 M. Grohe, D. Neuen, P. Schweitzer, and D. Wiebking. An improved isomorphism test for bounded-tree-width graphs. *ACM Trans. Algorithms*, 16(3):34:1–34:31, 2020.

11 M. Grohe, D. Wiebking, and D. Neuen. Isomorphism testing for graphs excluding small minors. In *61st IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 625–636, 2020.

12 J. L. Gross and T. W. Tucker. *Topological graph theory.* Dover, 2001.

13 J. L. Gross, J. Yellen, and P. Zhang. *Handbook of graph theory.* Chapman and Hall/CRC, 2013.

14 B. Grünbaum and G. C. Shephard. *Tilings and patterns.* Freeman, 1987.

15 Christoph M Hoffmann. Subcomplete generalizations of graph isomorphism. *Journal of Computer and System Sciences*, 25(3):332–359, 1982.

16 J. E. Hopcroft and R. Endre. Tarjan. A V log V algorithm for isomorphism of triconnected planar graphs. *J. Comput. Syst. Sci.*, 7(3):323–331, 1973.

17 J. E. Hopcroft and J. K. Wong. Linear time algorithm for isomorphism on planar graphs. In *Proc. 6th Annual ACM Symp. on Theory of Computing.*, 1974.

18 G. A. Jones and D. Singerman. Theory of maps on orientable surfaces. *Proceedings of the London Mathematical Society*, 3(2):273–307, 1978.

19 K. Kawarabayashi. Graph isomorphism for bounded genus graphs in linear time. *arXiv preprint*, 2015. `arXiv:1511.02460`.

20 K. Kawarabayashi, P. Klavík, B. Mohar, R. Nedela, and P. Zeman. Isomorphisms of maps on the sphere. In *Polytopes and Discrete Geometry*, volume 764 of *Contemporary Mathematics*, pages 125–147. American Mathematical Society, 2021.

21 K. Kawarabayashi and B. Mohar. Graph and map isomorphism and all polyhedral embeddings in linear time. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 471–480. ACM, 2008.

22 D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. Fixed-parameter tractable canonization and isomorphism test for graphs of bounded treewidth. *SIAM Journal on Computing*, 46(1):161–189, 2017.

23 E. M. Luks. Isomorphism of graphs of bounded valence can be tested in polynomial time. *Journal of computer and system sciences*, 25(1):42–65, 1982.

24 B. Mohar and C. Thomassen. *Graphs on surfaces*, volume 10. Johns Hopkins University Press, 2001.

25 R. Nedela and M. Škoviera. Exponents of orientable maps. *Proceedings of the London Mathematical Society*, 75(1):1–31, 1997.

26 D. Neuen. Hypergraph isomorphism for groups with restricted composition factors. In *47th International Colloquium on Automata, Languages, and Programming, (ICALP)*, volume 168 of *LIPIcs*, pages 88:1–88:19, 2020.

27 I. N. Ponomarenko. The isomorphism problem for classes of graphs closed under contraction. *Journal of Soviet Mathematics*, 55(2):1621–1643, 1991.

28 U. Schöning. Graph isomorphism is in the low hierarchy. *Journal of Computer and System Sciences*, 37(3):312–323, 1988.

29 Ondrej Šuch. Vertex-transitive maps on a torus. *Acta Math. Univ. Comenianae*, 80(1):1–30, 2011.

30 L. Weinberg. A simple and efficient algorithm for determining isomorphism of planar triply connected graphs. *IEEE Transactions on Circuit Theory*, 13(2):142–148, 1966.

31 H. Whitney. 2-isomorphic graphs. *American Journal of Mathematics*, 55(1):245–254, 1933.