

Algorithmic Aspects of Regular Graph Covers with Applications to Planar Graphs ☆, ☆☆

Jiří Fiala^a, Pavel Klavík^b, Jan Kratochvíl^a,
and Roman Nedela^c

^a Department of Applied Mathematics,
Faculty of Mathematics and Physics, Charles University,
Malostranské náměstí 25, 118 00 Prague, Czech Republic.
E-mails: {fiala,honza}@kam.mff.cuni.cz.

^b Computer Science Institute,
Faculty of Mathematics and Physics, Charles University,
Malostranské náměstí 25, 118 00 Prague, Czech Republic.
E-mail: klavik@iuuk.mff.cuni.cz.

^c Institute of Mathematics and Computer Science SAS
and Matej Bel University,
Ďumbierska 1, 974 11 Banská Bystrica, Slovak republic.
Email: nedela@savbb.sk.

☆ **Acknowledgment:** The first three authors are supported by the ESF Eurogiga project GraDR as GAČR GIG/11/E023, the fourth author by the ESF Eurogiga project GReGAS as the APVV project ESF-EC-0009-10 and by VEGA 1/0621/11. The first author is also supported by the project Kontakt LH12095 and the second author by GAČR 14-14179S. The second and the third authors are also supported by Charles University as GAUK 196213.

☆☆ The conference version of this paper appeared in ICALP 2014.

Abstract

A graph G *covers* a graph H if there exists a locally bijective homomorphism from G to H . We deal with *regular covers* in which this locally bijective homomorphism is prescribed by an action of a subgroup of $\text{Aut}(G)$. Regular covers have many applications in constructions and studies of big objects all over mathematics and computer science.

We study *computational aspects* of regular covers that have not been addressed before. The decision problem **REGULARCOVER** asks for two given graphs G and H whether G regularly covers H . When $|H| = 1$, this problem becomes Cayley graph recognition for which the complexity is still unresolved. Another special case arises for $|G| = |H|$ when it becomes the graph isomorphism problem. Therefore, we restrict ourselves to graph classes with polynomially solvable graph isomorphism.

Inspired by Negami, we apply the structural results used by Babai in the 1970's to study automorphism groups of graphs. Our main result is the following **FPT** meta-algorithm: Let \mathcal{C} be a class of graphs such that the structure of automorphism groups of 3-connected graphs in \mathcal{C} is simple. Then we can solve **REGULARCOVER** for \mathcal{C} -inputs G in time $\mathcal{O}^*(2^{e(H)/2})$ where $e(H)$ denotes the number of the edges of H . As one example of \mathcal{C} , this meta-algorithm applies to planar graphs. In comparison, testing general graph covers is known to be **NP**-complete for planar inputs G even for small fixed graphs H such as K_4 or K_5 . Most of our results also apply to general graphs, in particular the complete structural understanding of regular covers for 2-cuts.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 5 |
| 1.1 | Applications of Graph Coverings | 6 |
| 1.2 | Complexity Aspects | 8 |
| 1.3 | Our Results | 10 |
| 2 | Definitions and Preliminaries | 13 |
| 2.1 | Automorphisms and Groups | 13 |
| 2.2 | Graph Coverings | 15 |
| 2.3 | Fundamental Complexity Properties of Graph Coverings | 17 |
| 2.4 | Overview of the Main Steps | 21 |
| 3 | Structural Properties of Atoms | 25 |
| 3.1 | Block-trees and Their Automorphisms | 25 |
| 3.2 | Definition and Basic Properties of Atoms | 27 |
| 3.3 | Symmetry Types of Atoms | 31 |
| 3.4 | Automorphisms of Atoms | 33 |
| 4 | Graph Reductions and Quotient Expansions | 37 |
| 4.1 | Reducing Graphs Using Atoms | 38 |
| 4.2 | Quotients and Their Expansion | 45 |
| 5 | Meta-algorithm | 53 |

| | | |
|----------|--|-----------|
| 5.1 | Testing Expandability Using Dynamic Programming | 54 |
| 5.2 | Proof of The Main Theorem | 62 |
| 5.3 | More Details about Star Atoms and Their Lists | 64 |
| 6 | Applying the Meta-algorithm to Planar Graphs | 77 |
| 6.1 | Automorphism Groups of 3-connected Planar Graphs | 77 |
| 6.2 | Primitive Graphs and Atoms for Planar Graphs | 79 |
| 6.3 | Planar Graphs Satisfy (P0) to (P3) | 82 |
| 7 | Concluding Remarks | 85 |
| 7.1 | Possible Extensions of The Meta-algorithm | 85 |
| 7.2 | Open problems | 85 |

1

Introduction

The notion of *covering* originates in topology as a notion of local similarity of two topological surfaces. For instance, consider the unit circle and the real line. Globally, these two surfaces are not the same, they have different properties, different fundamental groups, etc. But when we restrict ourselves to a small part of the circle, it looks the same as a small part of the real line; more precisely the two surfaces are locally homeomorphic, and thus they share the local properties. The notion of covering formalizes this property of two surfaces being *locally the same*.

More precisely, suppose that we have two topological spaces: a big one G and a small one H . We say that G *covers* H if there exists a mapping called a *covering projection* $p : G \rightarrow H$ which locally preserves the structure of G . For instance, the mapping $p(x) = (\cos x, \sin x)$ from the real line to the unit circle is a covering projection. The existence of a covering projection ensures that G looks locally the same as H ; see Figure 1.1.

In this paper, we study coverings of graphs in a more restricting

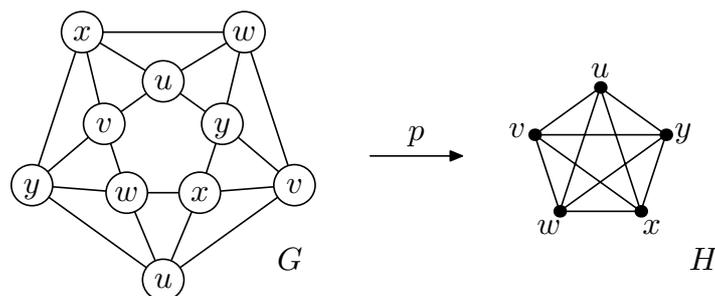


Figure 1.1: A covering projection p from a graph G to a graph H .

version called *regular covering*, for which the covering projection is described by an action of a group; see Section 2 for the formal definition. If G regularly covers H , then we say that H is a *quotient* of G .

1.1 Applications of Graph Coverings

Suppose that G covers H and we have some information about one of the objects. How much knowledge does translate to the other object? It turns out that quite a lot, and this makes covering a powerful technique with many diverse applications. The big advantage of regular coverings is that they can be efficiently described and many properties easily translate between the objects. We sketch some applications now.

Powerful Constructions. The reverse of covering called *lifting* can be applied to small objects in order to construct large objects of desired properties. For instance, the well-known Cayley graphs are large objects which can be described easily by a few elements of a group. Let G be a Cayley graph generated by elements g_1, \dots, g_e of a group Γ . The vertices of G correspond to the elements of Γ and the edges are described by actions of g_1, \dots, g_e on Γ by left multiplication; each g_i defines a permutation on Γ and we put edges along the cycles of this permutation. See Figure 1.2 for an example. Cayley graphs were originally invented to study the structure of groups [13].

In the language of coverings, every Cayley graph G with an involution-free generating set can be described as a lift of a one vertex graph H with e loops labeled g_1, \dots, g_e . Regular covers can be viewed as a generalization of Cayley graphs where the small graph H can con-

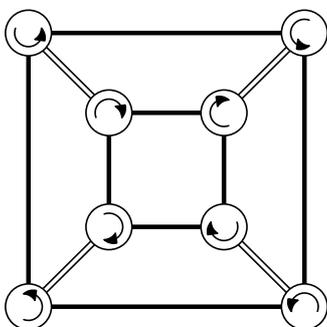


Figure 1.2: The Cayley graph of the dihedral group \mathbb{D}_4 generated by the 90° rotations (in black) and the reflection around the x -axis (in white).

1.1. Applications of Graph Coverings

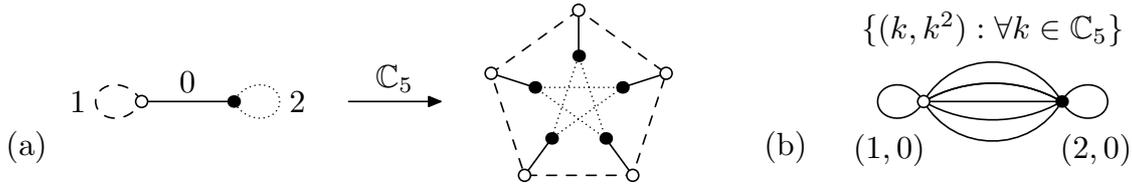


Figure 1.3: (a) A construction of the Petersen graph by lifting with the group \mathbb{C}_5 . (b) By lifting the described graph with the group \mathbb{C}_5^2 , we get the Hoffman-Singleton graph. The five parallel edges are labeled $(0, 0)$, $(1, 1)$, $(2, 4)$, $(3, 4)$ and $(4, 1)$.

tain more than one vertex. For example, the famous Petersen graph can be constructed as a lift of a two vertex graph H , see Figure 1.3a. These two vertices are necessary as it is known that Petersen graph is not a Cayley graph. Figure 1.3b shows a simple construction [35] of the Hoffman-Singleton graph [27] which is a 7-regular graph with 50 vertices. Notice that from this construction it is immediately apparent that the Hoffman-Singleton graph contains many induced copies of the Petersen graph. (In fact, it contains 525 copies of it.)

The Petersen and the Hoffman-Singleton graphs are extremal graphs for the *degree-diameter* problem: Given integers d and k , find a maximal graph G with diameter d and degree k . In general, the size of G is not known. Many currently best constructions are obtained using the covering techniques [36].

Further applications employ the fact that nowhere-zero flows, vertex and edge colorings, eigenvalues and other graph invariants lift along a covering projection. In history, two main applications are the solution of the Heawood map coloring problem [38, 23] and construction of arbitrary large highly symmetrical graphs [7].

Models of Local Computation. These and similar constructions have many practical applications in designing highly efficient computer networks [15, 1, 6, 10, 11, 12, 24, 42] since these networks can be efficiently described/constructed and have many strong properties. In particular, networks based on covers of simple graphs allow fast parallelization of computation as described e.g. in [9, 2, 3].

Simplifying Objects. Regular covering can be also applied in the opposite way, to project big objects onto smaller ones while preserving some properties. One way is to represent a class of objects satisfying some properties as quotients of the universal object of this prop-

erty. For instance, this was used in the study of arc-transitive cubic graphs [22], and the key point is that universal objects are much easier to work with. This idea is commonly used in fields such as Riemann surfaces [17] and theoretical physics [30].

1.2 Complexity Aspects

In the constructions we described, the covers are regular and satisfy additional algebraic properties. The reason is that regular covers are easier to describe. In this paper, we initiate the study of the computational complexity of regular covering.

Problem: REGULARCOVER
Input: Connected graphs G and H .
Output: Does G regularly cover H ?

Relations to Covers. This problem is closely related to the complexity of general covering which was widely studied before. We try to understand how much the additional algebraic structure changes the computational complexity. Study of the complexity of general covers was pioneered by Bodlaender [9] in the context of networks of processors in parallel computing, and for fixed target graph was first asked by Abello et al. [19]. The problem H -COVER asks for an input graph G whether it covers a fixed graph H . The general complexity is still unresolved but papers [31, 20] show that it is **NP**-complete for every r -regular graph H where $r \geq 3$. (For a survey of the complexity results, see [21].)

The complexity results concerning graph covers are mostly **NP**-complete. In our impression, the additional algebraic structure of regular graph covers makes the problem easier, as shown by the following two contrasting results. The problem H -COVER remains **NP**-complete for several small fixed graphs H (such as K_4 , K_5) even for planar inputs G [8]. On the other hand, our main result is that the problem REGULARCOVER is fixed-parameter tractable in the number of edges of H for every planar graph G and for every H .

Two additional problems of finding lifts and quotients, closely related to REGULARCOVER, are considered in Section 2.3.

Relations to Cayley Graphs and Graph Isomorphism. The notion of regular covers builds a bridge between two seemingly different problems. If the graph H consists of a single vertex, it corresponds to recognizing Cayley graphs which is still open; a polynomial-time algorithm is known only for circulant graphs [16]. When both graphs G and H have the same size, we get graph isomorphism testing. Our results are far from this but we believe that better understanding of regular covering can also shed new light on these famous problems.

Theoretical motivation for studying graph isomorphism is very similar to REGULARCOVER. For practical instances, one can solve the isomorphism problem very efficiently using various heuristics. But polynomial-time algorithm working for all graph is not known and it is very desirable to understand the complexity of graph isomorphism. It is known that testing graph isomorphism is equivalent to testing isomorphism of general mathematical structures [25]. The notion of isomorphism is widely used in mathematics when one wants to show that two seemingly different mathematical structures are the same. One proceeds by guessing a mapping and proving that this mapping is an isomorphism. The natural complexity question is whether there is a better way in which one algorithmically derives an isomorphism. Similarly, regular covering is a well-known mathematical notion which is algorithmically interesting and not understood.

Further, a regular covering is described by a semiregular subgroup of the automorphism group $\text{Aut}(G)$. Therefore it seems to be closely related to computation of $\text{Aut}(G)$ since one should have a good understanding of this group first to solve the regular covering problem. The problem of computing automorphism groups is known to be closely related to graph isomorphism.

Homomorphisms and CSP. Since regular covering is a restricted locally bijective homomorphism, we give an overview of complexity results concerning homomorphisms and general coverings. Hell and Nešetřil [26] studied the problem H -HOM which asks whether there exists a homomorphism between an input graph G and a fixed graph H . Their celebrated dichotomy result for simple graphs states that the problem H -HOM is polynomially solvable if H is bipartite and it is NP-complete otherwise. Homomorphisms can be described in the language of constraint satisfaction (CSP), and the famous dichotomy conjecture [18] claims that every CSP is either polynomially solvable,

or NP-complete.

1.3 Our Results

Let \mathcal{C} be a class of connected multigraphs. By \mathcal{C}/Γ we denote the class of all regular quotients of graphs of \mathcal{C} (note that $\mathcal{C} \subseteq \mathcal{C}/\Gamma$). For instance for \mathcal{C} equal to planar graphs, the class \mathcal{C}/Γ is – according to the Negami’s Theorem [37] – the class of projective planar graphs. We consider four properties of \mathcal{C} , for formal definitions see Section 2:

- (P0) The class \mathcal{C} is closed under taking subgraphs and under replacing connected components attached to 2-cuts by edges.
- (P1) The graph isomorphism problem is solvable in polynomial time for \mathcal{C} and \mathcal{C}/Γ .
- (P2) For a 3-connected graph $G \in \mathcal{C}$, the group $\text{Aut}(G)$ and all its semiregular subgroups Γ can be computed in polynomial time. Here by semiregularity, we mean that the action of Γ has no non-trivial stabilizers of the vertices.
- (P3) Let G and H be 3-connected graphs of \mathcal{C}/Γ , possibly with colored and directed edges. Let the vertices of G be further colored by $c(u), u \in V(G)$, and let H be equipped with a list $\mathfrak{L}(u)$ of possible colors for each vertex $u \in V(H)$ (the coloring is not necessarily proper). We can test in polynomial time whether there exists a color-compatible isomorphism $\pi : G \rightarrow H$, i.e. an isomorphism such that the colors and orientations of edges are preserved and for every $u \in V(G)$, we have $c(u) \in \mathfrak{L}(\pi(u))$. (The existence of such an isomorphism is denoted by $G \hookrightarrow H$.)

As we prove in Section 6, these four properties are tailored for the class of planar graphs. (But the proof of the property (P3) is non-trivial, based on the result of [32].) Negami’s Theorem [37] dealing with regular covers of planar graphs is one of the oldest results in topological graph theory; therefore we decided to start the study of computational complexity of REGULARCOVER for planar graphs. Our algorithm, however, applies to a wider class of graphs.

We use the complexity notation $f = \mathcal{O}^*(g)$ which omits polynomial factors. Our main result is the following FPT meta-algorithm:

Theorem 1.1. *Let \mathcal{C} be a class of graphs satisfying (P0) to (P3). Then there is an FPT algorithm for REGULARCOVER for \mathcal{C} -inputs G in the parameter $e(H)$, running in time $\mathcal{O}^*(2^{e(H)/2})$ where $e(H)$ is the number of edges of H .*

It is important that most of our results apply to general graphs. We wanted to generalize the result of Babai [4] which states that it is sufficient to solve graph isomorphism for 3-connected graphs. Our main goal was to understand how regular covering behaves with respect to vertex 1-cuts and 2-cuts. Concerning 1-cuts, regular covering behaves non-trivially only on the central block of G , so they are easy to deal with. But we show that regular covering can behave highly complex on 2-cuts. From structural point of view, we give a complete description of this behaviour. Algorithmically, we solve computation only partially and we need several other assumptions to get an efficient algorithm.

Planar graphs are very important and also well studied in connection to coverings. Negami's Theorem [37] dealing with regular covers of planar graphs is one of the oldest results in topological graph theory; therefore we decided to start the study of computational complexity of REGULARCOVER for planar graphs. In particular, our theory applies to planar graphs since they satisfy (P0) to (P3).

Corollary 1.2. *For a planar graph G , REGULARCOVER can be solved in time $\mathcal{O}^*(2^{e(H)/2})$.*

Our Approach. We sketch our approach. The meta-algorithm proceeds by a series of *reductions* replacing parts of the graphs by edges. These reductions are inspired by the approach of Negami [37] and turn out to follow the same lines as the reductions introduced by Babai for studying automorphism groups of planar graphs [4, 5]. Since the key properties of the automorphism groups are preserved by the reductions, computing automorphism groups can be reduced to computing them for 3-connected graphs [4]. In [29, 28], this is used to compute automorphism groups of planar graphs since the automorphism groups of 3-connected planar graphs are the automorphism groups of tilings of the sphere, and are well-understood.

The REGULARCOVER problem is more complicated, and we use the following novel approach. When the reductions reach a 3-connected graph, the natural next step is to compute all its quotients; there are

polynomially many of them according to (P2). What remains is the most difficult part: To test for each quotient whether it corresponds to H after unrolling the reductions. This process is called *expanding* and the issue here is that there may be exponentially many different ways to expand the graph, so we have to test in a clever way whether it is possible to reach H . Our algorithm consists of several subroutines, most of which we indeed can perform in polynomial time. Only one subroutine (finding a certain “generalized matching”) we have not been able to solve in polynomial time.

This slow subroutine can be avoided in some cases:

Corollary 1.3. *If the \mathcal{C} -graph G is 3-connected or if $k = |G|/|H|$ is odd, then the meta-algorithm of Theorem 1.1 can be modified to run in polynomial time.*

Structure. This paper is organized as follows. In Section 2, we introduce the formal notation used in this paper. In Section 3, we introduce *atoms* which are the key objects of this paper. In Section 4, we describe structural properties of reductions via atoms, and expansions of constructed quotient graphs. In Section 5, we use these structural properties to create the meta-algorithm of Theorem 1.1. Finally, in Section 6 we deal with specific properties of planar graphs and show that the class of planar graphs satisfies (P0) to (P3). In Conclusions, we describe open problems and possible extensions of our results. See Section 2.4 for more detailed overview of the main steps.

2

Definitions and Preliminaries

A multigraph G is a pair $(V(G), E(G))$ where $V(G)$ is a set of vertices and $E(G)$ is a multiset of edges. We denote $|V(G)|$ by $v(G)$ and $|E(G)|$ by $e(G)$. The graph can possibly contain parallel edges and loops, and each loop at u is incident twice with the vertex u . (So it contributes by two to the degree of u .) Each edge $e = uv$ gives rise to two half-edges, one attached to u and the other to v . We denote by $H(G)$ the collection of all half-edges. We denote $|H(G)|$ by $h(G)$ and clearly $h(G) = 2e(G)$. As quotients, we sometime obtain graphs containing half-edges with free ends (missing the opposite half-edges).

We consider graphs with colored edges and also with three different edge types (directed edges, undirected edges and a special type called halvable edges). It might seem strange to consider such general objects. But when we apply reductions, we replace parts of the graph by edges and the colors encode isomorphism classes of replaced parts. This allows the algorithm to work with smaller reduced graphs and deduce some structure of the original large graph. So even if the input graphs G and H are simple, the more complicated multigraphs are naturally constructed.

2.1 Automorphisms and Groups

Automorphisms. We state the definitions in a very general setting of multigraphs and half-edges. An *automorphism* π is fully described by a permutation $\pi_h : H(G) \rightarrow H(G)$ preserving edges and incidences between half-edges and vertices. Thus, π_h induces two permutations $\pi_v : V(G) \rightarrow V(G)$ and $\pi_e : E(G) \rightarrow E(G)$ connected together by the

Chapter 2. Definitions and Preliminaries

very natural property $\pi_e(uv) = \pi_v(u)\pi_v(v)$ for every $uv \in E(G)$. In most of situations, we omit subscripts and simply use $\pi(u)$ or $\pi(uv)$. In addition, when we work with colored graphs, we require that an automorphism preserves the colors.

Groups. We assume that the reader is familiar with basic properties of groups; otherwise see [39]. We denote groups by Greek letters as for instance Γ . We use the following notation for standard groups:

- \mathbb{S}_n for the symmetric group of all n -element permutations,
- \mathbb{C}_n for the cyclic group of integers modulo n ,
- \mathbb{D}_n for the dihedral group of the symmetries of a regular n -gon, and
- \mathbb{A}_n for the group of all even n -element permutations.

Automorphism Groups. Groups are quite often studied in the context of group actions, since their origin is in studying symmetries of mathematical objects. A group Γ *acts* on a set S in the following way. Each $g \in \Gamma$ permutes the elements of S , and the *action* is described by a mapping $\cdot : \Gamma \times S \rightarrow S$ usually satisfying further properties that arise naturally from the structure of S .

In the language of graphs, an example of such an action is the group of all automorphisms of G , denoted by $\text{Aut}(G)$. Each element $\pi \in \text{Aut}(G)$ acts on G , permutes its vertices, edges and half-edges while it preserves edges and incidences between the half-edges and the vertices.

The *orbit* $[v]$ of a vertex $v \in V(G)$ is the set of all vertices $\{\pi(v) \mid \pi \in \Gamma\}$, and the orbit $[e]$ of an edge $e \in E(G)$ is defined similarly as $\{\pi(e) \mid \pi \in \Gamma\}$. The *stabilizer* of x is the subgroup of all automorphisms which fix x . An action is called *semiregular* if it has no non-trivial (i.e., non-identity) stabilizers of both vertices and half-edges. Note that the stabilizer of an edge in a semiregular action may be non-trivial, since it may contain an involution transposing the two half-edges. We say that a group is *semiregular* if the associated action is semiregular.

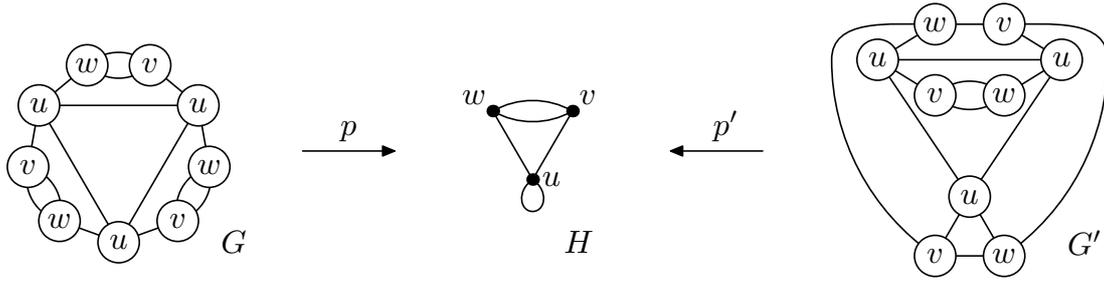


Figure 2.1: Two covers of H . The projections p_v and p'_v are written inside of the vertices, and the projections p_e and p'_e are omitted. Notice that each loop is realized by having two neighbors labeled the same, and parallel edges are realized by having multiple neighbors labeled the same. Also covering projections preserve degrees.

2.2 Graph Coverings

A graph G covers a graph H (or G is a *cover* of H) if there exists a locally bijective homomorphism p called a *covering projection*. A homomorphism p from G to H is given by a mapping $p_h : H(G) \rightarrow H(H)$ preserving edges and incidences between half-edges and vertices. It induces two mappings $p_v : V(G) \rightarrow V(H)$ and $p_e : E(G) \rightarrow E(H)$ such that $p_e(uv) = p_v(u)p_v(v)$ for every $uv \in E(G)$. The property to be local bijective states that for every vertex $u \in V(G)$ the mapping p_h restricted to the half-edges incident with u is a bijection. Figure 2.1 contains two examples of graph covers. Again, we mostly omit subscripts and just write $p(u)$ or $p(e)$. A *fiber* of a vertex $v \in V(H)$ is the set $p^{-1}(v)$, i.e., the set of all vertices $V(G)$ that are mapped to v , and similarly for fibers over edges and half-edges.

The Unique Walk Lifting Property. Let $uv \in E(H)$ be an edge which is not a loop. Then the set $p^{-1}(uv)$ corresponds to a perfect matching between the fibers $p^{-1}(u)$ and $p^{-1}(v)$. And if $uu \in E(H)$ is a loop, then the set $p^{-1}(uu)$ is a union of disjoint cycles which cover exactly $p^{-1}(u)$. Figure 2.2 shows examples. In general for a subgraph H' of H , the correspondence between H' and its preimages $p^{-1}(H')$ in G is called *lifting*.

Let W be a walk $u_0e_1u_1e_2 \dots e_nu_n$ in H . Then $p^{-1}(W)$ consists of k copies of W . Suppose that we fix as the start one vertex in the fiber $p^{-1}(u_0)$. Due to the local bijectiveness of p , there is exactly one edge incident with it which p maps to e_1 , so $p^{-1}(u_1)$ is now uniquely deter-

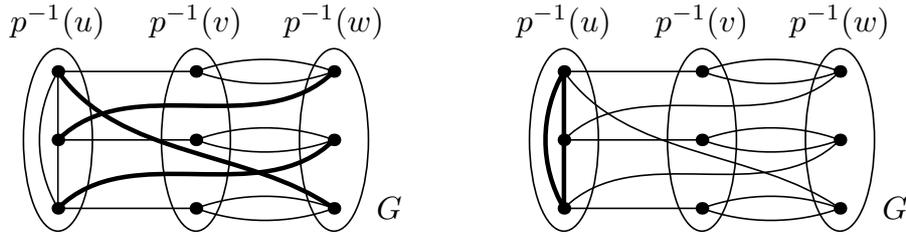


Figure 2.2: The graph G from Figure 2.1 depicted by fibers of p . On the left, we show that $p^{-1}(uw)$ gives a matching between the fibers of u and w . On the right, the loop around u gives a cycle in $p^{-1}(u)$.

mined, and so on for $p^{-1}(u_2)$ and the other vertices of W . When we proceed in this way, the rest of the walk is determined. This important property of every covering is called the *unique walk lifting property*.

We adopt the standard assumption that both G and H are connected. Then as a simple corollary we get that all fibers of p are of the same size. To see that, observe that a path in H is lifted to disjoint paths in G . For $u, v \in V(H)$, consider a path P between them. Then the paths in $p^{-1}(P)$ define a bijection between the fibers $p^{-1}(u)$ and $p^{-1}(v)$. In other words, $|G| = k|H|$ for some $k \in \mathbb{N}$ which is the size of each fiber, and we say that G is a *k-fold cover* of H .

Covering Transformation Groups. Every covering projection p defines a special subgroup of $\text{Aut}(G)$ called *covering transformation group* $\text{CT}(p)$. It consists of all automorphisms π which preserve the fibers of p , i.e., for every $u \in V(G)$, the vertices u and $\pi(u)$ belong to one fiber. Consider the graphs from Figure 2.1. For the graph G , we have $\text{Aut}(G) = \mathbb{D}_3$ and $\text{CT}(p) = \mathbb{C}_3$. And G' has $\text{Aut}(G') = \mathbb{C}_2$ but $\text{CT}(p')$ is trivial; and we note that it is often the case that a covering projection p has only one fiber-preserving automorphism, the trivial one.

Now suppose that $\pi \in \text{CT}(p)$. Observe that a single choice of the image $\pi(u)$ of one vertex $u \in V(G)$ fully determines π . This follows from the unique walk lifting property. Let $v \in V(G)$ and consider some path $P_{u,v}$ connecting u and v in G . This path corresponds to a path $P = p(P_{u,v})$ in H . Now we lift P and according to the unique walk lifting property, there exists a unique path $P_{\pi(u),x}$ which starts in $\pi(u)$. But since π is an automorphism and it maps $P_{u,v}$ to $P_{\pi(u),x}$, then x has to be equal $\pi(v)$. In other words, we just proved that $\text{CT}(p)$ is semiregular.

2.3. Fundamental Complexity Properties of Graph Coverings

Regular Coverings. We want to consider coverings which are highly symmetrical. For examples from Figure 2.1, the covering p is more symmetric than p' . The size of $\text{CT}(p)$ is a good measure of symmetry of the covering p . Since $\text{CT}(p)$ is semiregular, it easily follows that $|\text{CT}(p)| \leq k$ for any k -fold covering p . A covering p is *regular* if $|\text{CT}(p)| = k$. In Figure 2.1, the covering p is regular since $|\text{CT}(p)| = 3$, and the covering p' is not regular since $|\text{CT}(p')| = 1$.

We use the following definition of regular covering. Let Γ be any semiregular subgroup of $\text{Aut}(G)$. It defines a graph G/Γ called a *regular quotient* (or simply *quotient*) of G as follows: The vertices of G/Γ are the orbits of the action Γ on $V(G)$, the half-edges of G/Γ are the orbits of the action Γ on $H(G)$. A vertex-orbit $[v]$ is incident with a half-edge-orbit $[h]$ if and only if the vertices of $[v]$ are incident with the half-edges of $[h]$. (Because the action of Γ is semiregular, each vertex of $[v]$ is incident with exactly one half-edge of $[h]$, so this is well defined.) We naturally construct $p : G \rightarrow G/\Gamma$ by mapping the vertices to its vertex-orbits and half-edges to its half-edge-orbits. Concerning an edge $e \in E(G)$, it is mapped to an edge of G/Γ if the two half-edges belong to different half-edge-orbits of Γ . If they belong to the same half-edge-orbit, it corresponds to a standalone half-edge of G/Γ .

Since Γ acts semiregularly on G , one can prove that p is a $|\Gamma|$ -fold regular covering with $\text{CT}(p) = \Gamma$. For the graphs G and H of Figure 2.1, we get $H \cong G/\Gamma$ for $\Gamma \cong \mathbb{C}_3$ which “rotates the cycle by three vertices”. As a further example, Figure 2.3 geometrically depicts all quotients of the cube graph.

2.3 Fundamental Complexity Properties of Graph Coverings

We establish fundamental complexity properties of regular covering and also general covering. Our goal is to highlight similarities with the graph isomorphism problem. Also we discuss other variants of the REGULARCOVER problem.

Belonging to NP. The general H -COVER problem is clearly in NP since one can just test in polynomial time whether a given mapping is a locally bijective homomorphism. Not so obviously, the same holds for the REGULARCOVER problem.

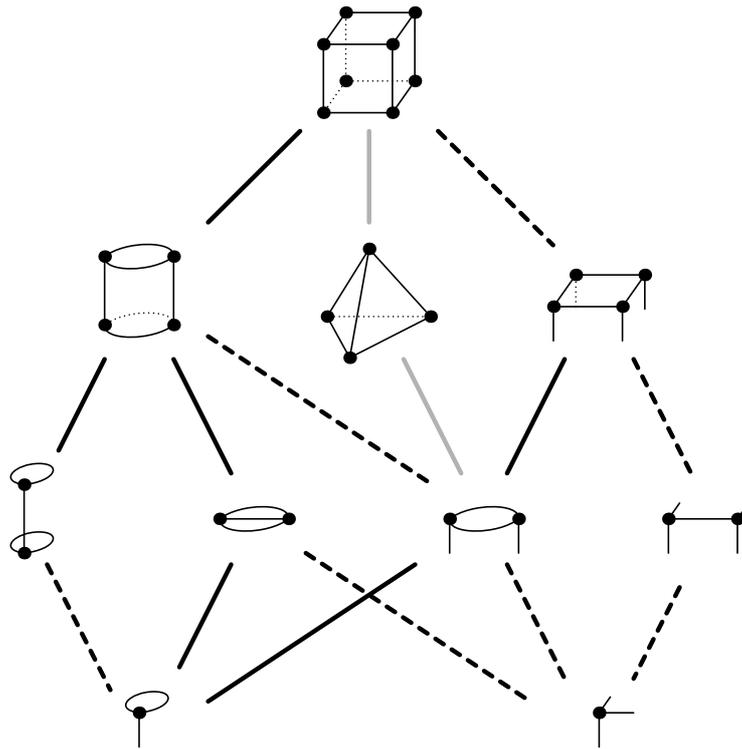


Figure 2.3: The Hasse diagram of all quotients of the cube graph depicted in a geometric way. When semiregular actions fix edges, the quotients contain half-edges. The quotients connected by bold edges are obtained by 180 degree rotations. The quotients connected by dashed edges are obtained by reflections. The tetrahedron is obtained by the antipodal symmetry of the cube, and its quotient is obtained by a 180 degree rotation with the axis going through the centers of two non-incident edges of the tetrahedron.

Lemma 2.1. *The problem REGULARCOVER is in NP.*

Proof. One just needs to use a suitable definition of regular covering. As stated above, G regularly covers H , if and only if there exists a semiregular subgroup Γ of $\text{Aut}(G)$ such that $G/\Gamma \cong H$. As the certificate, we just give k permutations, one for each element of Γ , and the isomorphism between G/Γ and H . We can easily check whether these k permutations define a group Γ , and whether Γ acts semiregularly on G . Further, the given isomorphism allows to check whether the constructed G/Γ is isomorphic to H . Clearly, this certificate is polynomially large and can be verified in polynomial time. \square

One can prove even a stronger result:

2.3. Fundamental Complexity Properties of Graph Coverings

Lemma 2.2. *For a mapping $p : G \rightarrow H$, we can test whether it is a regular covering in polynomial time.*

Proof. Testing whether p is a covering can clearly be done in polynomial time. It remains to test regularity. Choose an arbitrary spanning tree T of H . Since p is a covering, then $p^{-1}(T)$ is a disjoint union of k isomorphic copies T_1, \dots, T_k of T . We number the vertices of the fibers according to the spanning trees, i.e., $p^{-1}(v) = \{v_1, \dots, v_k\}$ such that $v_i \in T_i$. This induces a numbering of the half-edges of each fiber over a half-edge of $H(H)$, following the incidences between half-edges and vertices. For every half-edge $h \notin H(T)$, we define a permutation σ_h of $\{1, \dots, k\}$ taking i to j if there is a half-edge h in $p^{-1}(h)$ such that the edge $u_i v_j$ corresponds to h .

It remains to test whether the size of the group Θ generated by all σ_h , where $h \notin H(T)$, is of size exactly k . From the theory of permutation groups, since G is assumed to be connected, it follows that the action of Θ is transitive. Therefore its size is at least k , and the action is regular if and only if it is exactly k . \square

The constructed permutations σ_h associated with p are known in the literature [23] as *permutation voltage assignments* associated with p .

Other Variants. In the REGULARCOVER problem, the input gives two graphs G and H and we ask for an existence of a regular covering from G to H . There are two other reasonable variants of this problem we discuss now. The input can specify only one of the two graphs and ask for existence of the other graph of, say, a given size.

First, suppose that only H is given and we ask whether a k -fold cover G of H exists. This is called *lifting* and the answer is always positive. The theory of covering describes a technique called voltage assignment which can be applied to generate all k -folds G . We do not deal with lifting in this paper, but there are nevertheless many interesting computational questions with important applications. For instance, one can try to generate efficiently all lifts up to isomorphism; this is not trivial since different voltage assignments might lead to isomorphic graphs. Also, one may ask for existence of a lift with some additional properties.

The other variant gives only G and asks for existence of a quotient H which is regularly covered by G and $|G| = k|H|$. This problem is **NP**-complete even for fixed $k = 2$, proved in a different language by Lubiw [33]. Lubiw shows that testing existence of a fixed-point free involutory automorphism is **NP**-complete which is equivalent to existence of a half-quotient H . We sketch here reduction from 3-SAT. Each variable is represented by an even cycle attached to the rest of the graph. Each cycle has two possible regular quotients, either a cycle of half length (obtained by the 180° rotation), or a path of half length with attached half-edges (obtained by a reflection through opposite edges). Each of these quotients represents one truth assignment of the corresponding variable. To distinguish variables, distinct gadgets are attached to the cycles. These variable gadgets are attached to clause gadgets. Naturally, one can construct a quotient of the clause gadget if and only if at least one literal is satisfied.

One should ask whether this reduction also implies **NP**-completeness for the **REGULARCOVER** problem. Since the input gives also a graph H , one can decode the assignment of the variables from it, and thus this reduction does not work. We conjecture that no similar reduction with a fixed k can be constructed for **REGULARCOVER** since we believe that for a fixed k the problem of counting the number of regular coverings between G and H can be solved using polynomially-many instances of **REGULARCOVER**. In complexity theory, it is believed that the counting version of no **NP**-complete problem satisfies this. Similar evidence was used by Mathon [34] to show that graph isomorphism is unlikely **NP**-complete, and as a work in progress we believe that a similar argument can be applied to **REGULARCOVER**.

The results of this paper also show that the reduction of Lubiw cannot be modified for planar inputs G . Our algorithmic and structural insights allow an efficient enumeration of all quotients H of a given planar graph G . On the other hand, the hardness result of Lubiw states that to solve the **REGULARCOVER** problem in general, one has to work with both graphs G and H from beginning. Our algorithm starts only with G and tries to match its quotients to H only in the end. Nevertheless, some modifications in this directions, not necessary for planar graphs, would be possible.

2.4 Overview of the Main Steps

We now give a quick overview of the paper.

The main idea is the following. If the input graph G is 3-connected, using our assumptions the `REGULARCOVER` problem is trivially solvable. Otherwise, we proceed by a series of reductions, replacing parts of the graph by edges, essentially forgetting details of the graph. We end-up with a primitive graph which is either 3-connected, or very simple (a cycle or K_2). The reductions are done in such a way that no essential information of semiregular actions is lost.

Inspired by Negami [37] and Babai [5], we introduce in Section 3 the most important definition of an *atom*. Atoms are inclusion-minimal subgraphs which cannot be further simplified and are essentially 3-connected. Our strategy for reductions is to detect the atoms and replace them by edges. In this process, we remove details from the graph but preserve its overall structure. Our definition of atoms is quite technical, dealing with many details necessary for the next sections.

When the graph G is not 3-connected, we consider its block-tree. The central block plays the key role in every regular covering projection. The reason is that the covering behaves non-trivially only on this central block; the remaining blocks are isomorphically preserved in H . Therefore the atoms are defined with respect to the central block. We distinguish three types of atoms:

- *Proper atoms* are inclusion-minimal subgraphs separated by a 2-cut inside a block.
- *Dipoles* are formed by the sets of all parallel edges joining two vertices.
- *Block atoms* are blocks which are leaves of the block-tree, or stars of all pendant edges attached to a vertex. The central block is never a block atom.

In Section 4, we deal with two transformations of graphs called *reduction* and *expansion*. The graph G is reduced by replacing its atoms by edges; for proper atoms and dipoles inside the blocks, for block atoms by pendant edges. Further these edges are colored according to isomorphism classes of the atoms. This way the reduction omits unimportant details from the graph but its key structure is preserved.

Chapter 2. Definitions and Preliminaries

We apply a series of reductions $G = G_0, \dots, G_r$ till we obtain a graph G_r called *primitive* which contains no atoms. We show that the reduction preserves essentially the automorphism group. More precisely, $\text{Aut}(G_i)$ is a factor-group of $\text{Aut}(G_{i-1})$; the action of $\text{Aut}(G_i)$ inside the atoms is lost by the factorization.

The other transformation called expansion is applied to the quotient graphs. The goal of expansion is to revert the reduction, so it replaces colored edges back by atoms. To do this, we have to understand how regular covering behaves with respect to atoms. Inspired by Negami [37], we show that each proper atom/dipole has three possible types of quotients that we call an *edge-quotient*, a *loop-quotient* and a *half-quotient*. The edge-quotient and the loop-quotient are uniquely determined but an atom may have many non-isomorphic half-quotients.

When the primitive graph G_r is reached, all semiregular subgroups Γ_r of $\text{Aut}(G_r)$ are computed and for each one a quotient $H_r = G_r/\Gamma_r$ is constructed. Our goal is to understand all graphs H_0 to which H_r can be expanded, as depicted in the following diagram:

$$\begin{array}{ccccccccccc}
 G_0 & \xrightarrow{\text{red.}} & G_1 & \xrightarrow{\text{red.}} & \cdots & \xrightarrow{\text{red.}} & G_i & \xrightarrow{\text{red.}} & G_{i+1} & \xrightarrow{\text{red.}} & \cdots & \xrightarrow{\text{red.}} & G_r \\
 \Gamma_0 \downarrow & & \Gamma_1 \downarrow & & & & \Gamma_i \downarrow & & \Gamma_{i+1} \downarrow & & & & \Gamma_r \downarrow \\
 H_0 & \xleftarrow{\text{exp.}} & H_1 & \xleftarrow{\text{exp.}} & \cdots & \xleftarrow{\text{exp.}} & H_i & \xleftarrow{\text{exp.}} & H_{i+1} & \xleftarrow{\text{exp.}} & \cdots & \xleftarrow{\text{exp.}} & H_r
 \end{array}$$

The constructed quotients contain colored edges, loops and half-edges corresponding to atoms. Each half-edge in H_r is created from a halvable edge if an automorphism of Γ_r fixes this halvable edge and exchanges its endpoints. Roughly speaking it corresponds to cutting the edge in half. We show that every possible expansion of a quotient H_{i-1} from H_i can be constructed by replacing the colored edges by the edge-quotients of the atoms, the colored loops by the loop-quotients and the color half-edges by some choices of half-quotients. This gives the complete structural description of all graphs which can be reached from H_r by expansion. Half-edges of H_{i-1} can arise only in expansions of half-edges of H_i .

In Section 5, we describe the meta-algorithm itself. From algorithmic point of view, the key difficulty arises from the fact that a graph H_r can have exponentially many pairwise non-isomorphic expansions H_0 . Therefore we cannot test all of them and we proceed in the opposite way. We start with the graph H and try to reach H_r by a series

2.4. Overview of the Main Steps

of reductions. But here the reductions are non-deterministic since a part of the graph H can correspond to many different subgraphs of G . Therefore, we keep lists of possible correspondences when we replace atoms in H . We proceed with the reductions and compute further lists using dynamic programming. There is only one slow subroutine which takes time $\mathcal{O}^*(2^{e(H)/2})$ which we describe in detail in Section 5.3.

In Section 6 we deal with specific properties of planar graphs and show that the meta-algorithm applies to them. It is a key observation that the REGULARCOVER problem is trivially solvable for 3-connected inputs G since the automorphism group $\text{Aut}(G)$ is spherical; it is either cyclic, dihedral or a subgroup of one of the three special groups. One can just enumerate all quotients of G and test graph isomorphism to H . The reason is that 3-connected planar graphs and their quotients behave geometrically. On the other hand, dealing with general planar graphs is rather non-trivial, since the geometry of the sphere is lost and it requires all the theory built in this paper. We establish that planar graphs satisfy the conditions (P0) to (P3) where especially the proof for (P3) is not straightforward.

3

Structural Properties of Atoms

In this section, we introduce special inclusion-minimal subgraphs of G called atoms. We show their structural properties such as that they behave nicely with respect to any covering projection.

3.1 Block-trees and Their Automorphisms

The *block-tree* T of G is a tree defined as follows. Consider all articulations in G and all maximal 2-connected subgraphs which we call *blocks* (with bridge-edges also counted as blocks). The block-tree T is the incidence graph between the articulations and the blocks. For an example, see Figure 3.1.

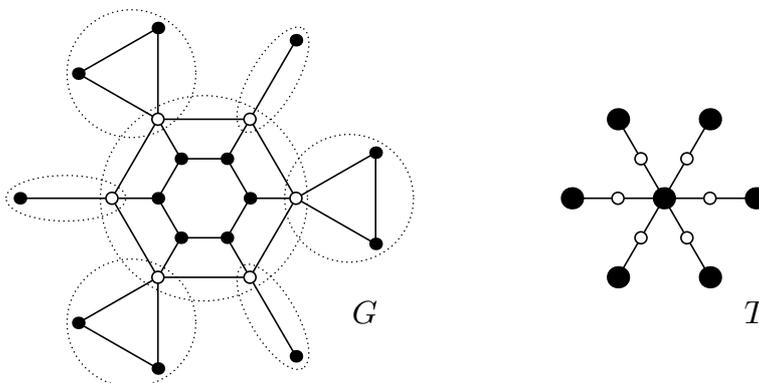


Figure 3.1: On the left, an example graph G with denoted blocks. On the right, the corresponding block-tree T is depicted. The white vertices correspond to the articulations and the big black vertices correspond to the blocks.

Chapter 3. Structural Properties of Atoms

There is the following well-known connection between $\text{Aut}(G)$ and $\text{Aut}(T)$:

Lemma 3.1. *Every automorphism $\pi \in \text{Aut}(G)$ induces an automorphism $\pi' \in \text{Aut}(T)$.*

Proof. First, observe that every automorphism π of G maps the articulations to the articulations and the blocks to the blocks which gives the induced mapping π' . It remains to show that π' is an automorphism of T . Let a be an articulation adjacent to a block B in the tree. Then a is contained in B . Therefore $\pi'(a)$ is contained in $\pi'(B)$ and vice versa, which implies that π' is an automorphism of the block-tree T . \square

We note that there is no direct relation between the structure of $\text{Aut}(G)$ and $\text{Aut}(T)$. First, $\text{Aut}(T)$ may contain some additional automorphisms not induced by anything in $\text{Aut}(G)$. Second, several distinct automorphisms of $\text{Aut}(G)$ may induce the same automorphism of $\text{Aut}(T)$. For example in Figure 3.1, $\text{Aut}(G) \cong \mathbb{D}_3 \times \mathbb{C}_2^3$ and $\text{Aut}(T) \cong \mathbb{S}_6$.

The Central Block. The *center* of a graph is a subset of the vertices which minimize the maximum distance to all vertices of the graph. For a tree, its center is either the central vertex or the central pair of vertices of a longest path, depending on the parity of its length. Every automorphism of a graph preserves its center.

Lemma 3.2. *If G has a non-trivial semiregular automorphism, then G has a central block.*

Proof. For the block-tree T , all leaves are blocks, so each longest path is of an even length. Therefore $\text{Aut}(T)$ preserves the central vertex. The central vertex can be either a *central articulation*, or a *central block*. If the central vertex is an articulation u , then every automorphism fixes u which contradicts the assumptions. \square

In the following, we shall assume that T contains a central block. We orient edges of the block-tree T towards the central block; so the block-tree becomes rooted. A *subtree* of the block-tree is defined by any vertex different from the central block acting as *root* and by all its descendants.

3.2. Definition and Basic Properties of Atoms

Let u be an articulation contained in the central block. By R_u we denote the subtree attached to the central block at u .

Lemma 3.3. *Let Γ be a semiregular subgroup of $\text{Aut}(G)$. If u and v are two articulations of the central block and of the same orbit of Γ , then $R_u \cong R_v$. Moreover there is a unique $\pi \in \Gamma$ which maps R_u to R_v .*

Proof. Notice that either $R_u = R_v$, or $R_u \cap R_v = \emptyset$. Since u and v are in the same orbit of Γ , there exists $\pi \in \Gamma$ such that $\pi(u) = v$. Consequently $\pi(R_u) = R_v$. Suppose that there exist $\pi, \sigma \in \Gamma$ such that $\pi(R_u) = \sigma(R_u) = R_v$. Then $\pi \cdot \sigma^{-1}$ is an automorphism of Γ fixing u . Since Γ is semiregular, $\pi = \sigma$. \square

3.2 Definition and Basic Properties of Atoms

Let u and v be two distinct vertices of degree at least three joined by at least two parallel edges. Then the subgraph induced by u and v is called a *dipole*. Let B be one block of G , so B is a 2-connected graph. Two vertices u and v form a *2-cut* $U = \{u, v\}$ if $B \setminus U$ is disconnected. We say that a 2-cut U is *non-trivial* if $\deg(u) \geq 3$ and $\deg(v) \geq 3$.

Lemma 3.4. *Let U be a 2-cut and let C be a component of $B \setminus U$. Then U is uniquely determined by C .*

Proof. If C is a component of $B \setminus U$, then U has to be the set of all neighbors of C in B . Otherwise B would not be 2-connected, or C would not be a component of $B \setminus U$. \square

The Definition. We first define a set \mathcal{P} of subgraphs of G which we call *parts*:

- A *block part* is a subgraph non-isomorphic to K_2 induced by the blocks of a subtree of the block-tree.
- A *proper part* is a subgraph S of G defined by a non-trivial 2-cut U of a block B not containing the central block. The subgraph S consists of a connected component C of $G \setminus U$ together with u and v and all edges between $\{u, v\}$ and C .
- A *dipole part* is any dipole.

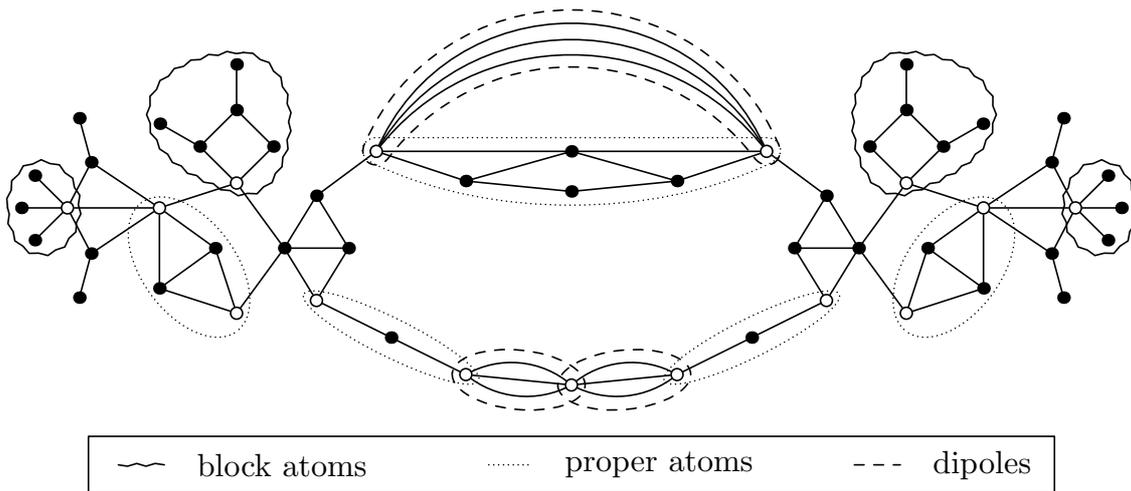


Figure 3.2: An example of a graph with denoted atoms. The white vertices belong to the boundary of some atom, possibly several of them.

The inclusion-minimal elements of \mathcal{P} are called *atoms*. We distinguish *block atoms*, *proper atoms* and *dipoles* according to the type of the defining part. Block atoms are either pendant stars, or pendant blocks possibly with single pendant edges attached to it. Also each proper atom or dipole is a subgraph of a block. For an example, see Figure 3.2.

We use topological notation to denote the *boundary* ∂A and the *interior* $\overset{\circ}{A}$ of an atom A . If A is a dipole, we set $\partial A = V(A)$. If A is a proper or block atom, we put ∂A equal the set of vertices of A which are incident with an edge not contained in A . For the interior, we use the standard topological definition $\overset{\circ}{A} = A \setminus \partial A$ where we only remove the vertices ∂A , the edges adjacent to ∂A are kept.

Note that $|\partial A| = 1$ for any block atom A , and $|\partial A| = 2$ for a proper atom or dipole A . The interior of a dipole is a set of free edges. We note that dipoles are automatically atoms and they are exactly the atoms with no vertices in their interiors. Observe for a proper atom A that the vertices of ∂A are exactly the vertices $\{u, v\}$ of the non-trivial 2-cut used in the definition of proper parts. Also the vertices of ∂A of a proper atom are never adjacent. Further, no block or proper atom contains parallel edges; otherwise a dipole would be its subgraph.

Properties of Atoms. Our goal is to replace atoms by edges, and so it is important to know that the atoms cannot overlap too much. The reader can see in Figure 3.2 that the atoms only share their boundaries. This is true in general, and we are going to prove it in two steps now.

3.2. Definition and Basic Properties of Atoms

Lemma 3.5. *The interiors of atoms are pairwise disjoint.*

Proof. For contradiction, let A and A' be two distinct atoms with non-empty intersections of \mathring{A} and \mathring{A}' . First suppose that A is a block item. Then A corresponds to a subtree of the block-tree which is attached by an articulation u to the rest of the graph. If A' is a block atom then it corresponds to some subtree, and we can derive that $A \subseteq A'$ or $A' \subseteq A$. And if A' is proper atom or dipole, then it is a subgraph of a block, and thus subgraph of A . In both cases, we get contradiction with minimality. Similarly, if one atom is a dipole, we can easily argue contradiction with minimality.

The last case is that both A and A' are proper atoms. Since the interiors are connected and the boundaries are defined as neighbors of the interiors, it follows that both $W' = A \cap \partial A'$ and $W = A' \cap \partial A$ are nonempty. We have two cases according to the sizes of these intersections depicted in Figure 3.3.

If $|W| = |W'| = 1$, then $W \cup W'$ is a 2-cut separating $\mathring{A} \cap \mathring{A}'$ which contradicts minimality of A and A' . And if, without loss of generality, $|W| = 2$, then there is no edge between $\mathring{A} \setminus (\mathring{A}' \cup W')$ and the remainder of the graph $G \setminus (\mathring{A} \cup \mathring{A}')$. Therefore, $\mathring{A} \setminus (\mathring{A}' \cup W')$ is separated by a 2-cut W' which again contradicts minimality of A . We note that in both cases the constructed 2-cut is non-trivial since it is formed by vertices of non-trivial cuts ∂A and $\partial A'$. \square

Next we show a stronger version of the previous lemma which states that two atoms can intersect only in their boundaries.

Lemma 3.6. *Let A and A' be two atoms. Then $A \cap A' = \partial A \cap \partial A'$.*

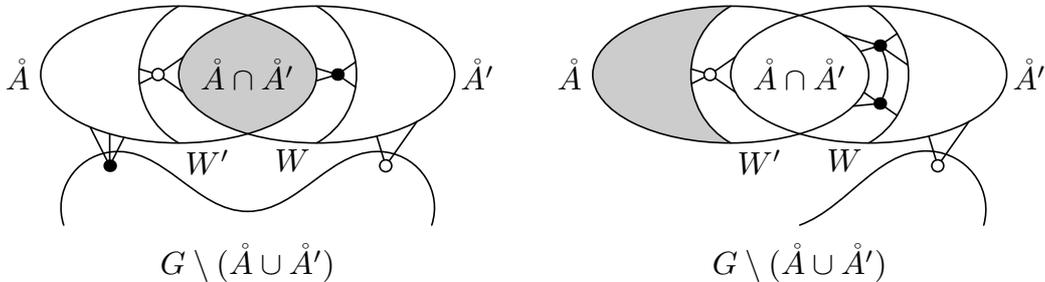


Figure 3.3: We depict the vertices of ∂A in black and the vertices of $\partial A'$ in white. In both cases, we found a subset of A belonging to \mathcal{P} (its interior is highlighted in gray).

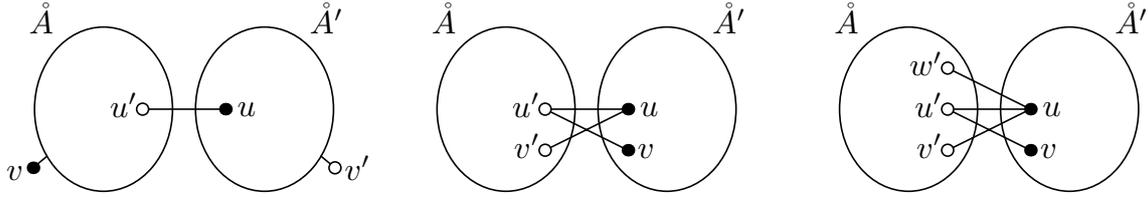


Figure 3.4: An illustration of the main steps of the proof.

Proof. We already know from Lemma 3.5 that $\mathring{A} \cap \mathring{A}' = \emptyset$. It remains to argue that, say, $\partial A \cap \mathring{A}' = \emptyset$. If A is a block atom, then ∂A is the articulation separating A . No atom can contain this articulation as its interior. Similarly, if A' is a block atom, then A has to be contained in the interior of A' or vice versa which contradicts minimality.

Let $\partial A = \{u, v\}$ and $\partial A' = \{u', v'\}$. First we deal with dipoles. The situation where A' is a dipole is trivial. And if A is a dipole with $u \in \mathring{A}'$, then either $v \in A$ which contradicts minimality of A , or $\partial A'$ is not a 2-cut. It remains to deal with both A and A' being proper atoms. Recall that in such a case ∂A is defined as neighbors of \mathring{A} in G , and that $\partial A'$ are neighbors of \mathring{A}' in G .

The proof is illustrated in Figure 3.4. Suppose for contradiction that $\mathring{A} \cap \partial A' \neq \emptyset$ and let $u' \in \mathring{A}$. Since u' has at least one neighbor in \mathring{A}' , then without loss of generality $u \in \mathring{A}'$ and $uu' \in E(G)$. Since A is a proper atom, the set $\{u', v\}$ is not a 2-cut, so there is another neighbor of u in \mathring{A} , which has to be equal v' . Symmetrically, u' has another neighbor in \mathring{A}' which is v . So $\partial A \subseteq \mathring{A}'$ and $\partial A' \subseteq \mathring{A}$. If $\partial A = \mathring{A}'$ and $\partial A' = \mathring{A}$, the graph is K_4 (since the minimal degree of cut-vertices is three) which contradicts existence of 2-cuts and atoms. And if for example $\mathring{A} \neq \partial A'$, then $\partial A'$ does not cut a subset of \mathring{A} , so there has to be a third neighbor w' of \mathring{A}' , which contradicts that $\partial A'$ cuts \mathring{A}' from the rest of the graph. \square

Connectivity of Atoms. We call a graph *essentially 3-connected* if it is a 3-connected graphs with possibly single pendant edges attached to it. For instance, every block atom is essentially 3-connected. A proper A might not be essentially 3-connected. Let $\partial A = \{u, v\}$. We define A^+ as A with the additional edge uv . Notice that the property (P0) ensures that A^+ belongs to \mathcal{C} . It is easy to see that A^+ is essentially 3-connected graph. Additionally, we put $A^+ = A$ for a block atom or dipole.

Lemma 3.7. *Let A be an essentially 3-connected graph, and we construct B from A by removing the single pendant edges of A . Then $\text{Aut}(A)$ is a subgroup of $\text{Aut}(B)$.*

Proof. These pendant single edges behave like markers, giving a 2-partition of $V(G)$ which $\text{Aut}(A)$ has to preserve. \square

Further, if $\text{Aut}(B)$ is of polynomial size, we can easily check which permutations preserve this 2-partition, and thus give $\text{Aut}(A)$. Also, similar relation holds for any group Γ acting on B and its subgroup Γ' preserving the 2-partition.

It is important that we can code the 2-partition by coloring the vertices of B , and work with such colored 3-connected graph using (P2) and (P3).

3.3 Symmetry Types of Atoms

We distinguish three symmetry types of atoms which describe how symmetric each atom is. When such an atom is reduced, we replace it by an edge carrying the type. Therefore we have to use multigraphs with three edge types: *halvable edges*, *undirected edges* and *directed edges*. We consider only the automorphisms which preserve these edge types and indeed the orientation of directed edges.

Let A be a proper atom or dipole with $\partial A = \{u, v\}$. Then we distinguish the following three symmetry types, see Figure 3.5:

- *The halvable atom.* There exists an semiregular involutory automorphism π which exchanges u and v . More precisely, the auto-

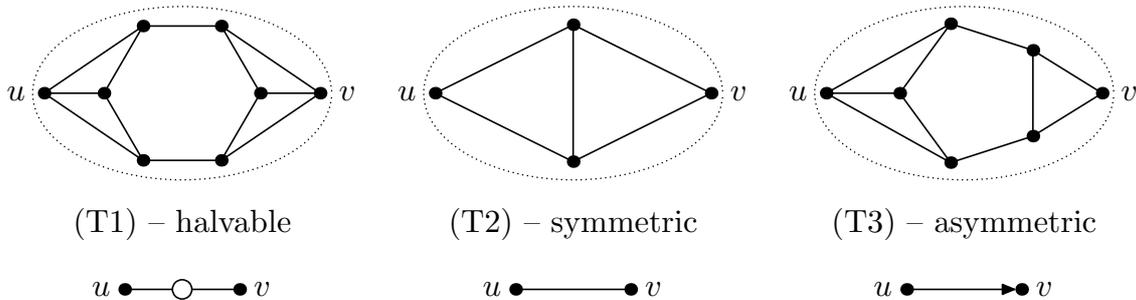


Figure 3.5: The three types of atoms and the corresponding edge types which we use in the reduction.

Chapter 3. Structural Properties of Atoms

morphism π fixes no vertices and no edges with an exception of some halvable edges.

- *The symmetric atom.* The atom is not halvable, but there exists an automorphism which exchanges u and v .
- *The asymmetric atom.* The atom which is neither halvable nor symmetric.

If A is a block atom, then it is by definition symmetric.

Lemma 3.8. *For a given dipole A , it is possible to determine its type in polynomial time.*

Proof. The type depends only on the quantity of distinguished types of the parallel edges. We have directed edges from u to v , directed edges from v to u , undirected edges and halvable edges. We call a dipole *balanced* if the number of directed edges in the both directions is the same. Observe that:

- The dipole A is halvable if and only if it is balanced and has an even number of undirected edges.
- The dipole A is symmetric if and only if it is balanced and has an odd number of undirected edges.
- The dipole A is asymmetric if and only if it is unbalanced.

This clearly can be tested in polynomial time. □

Lemma 3.9. *For a given proper atom A of \mathcal{C} satisfying (P2), it is possible to determine its type in polynomial time.*

Proof. Let $\partial A = \{u, v\}$. Recall that A^+ is an essentially 3-connected graph. Let B be the 3-connected graph created from A^+ by removing pendant edges, where existence of pendant edges is coded by colors of $V(B)$. Using (P3), we can check whether there is a color-preserving automorphism exchanging u to v as follows, see Figure 3.6. We take two copies of B . In one copy, we color u by a special color, and v by another special color. In the other copy, we swap the colors of u and v . Using (P3) on these two copies, we can check whether there is an automorphism which exchanges u and v . If not, then A is asymmetric. If yes, we check whether A is symmetric or halvable.

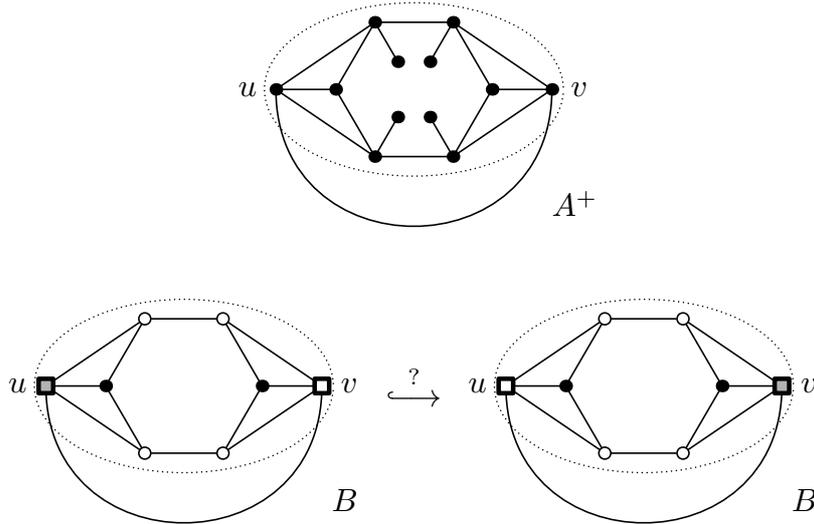


Figure 3.6: For the depicted atom A , we test using (P3) whether $B \leftrightarrow B$. In this case yes, so A is either symmetric, or halvable.

Using (P2), we generate polynomially many semiregular involutions of order two acting on B . For each semiregular involution, we check whether it transposes u to v , and whether it preserves the colors of $V(B)$ coding pendant edges. If such a semiregular involution exists, then A is halvable, otherwise it is just symmetric. \square

3.4 Automorphisms of Atoms

We start with a simple lemma which states how automorphisms behave with respect to atoms.

Lemma 3.10. *Let A be an atom and let $\pi \in \text{Aut}(G)$. Then the following holds:*

- (a) *The image $\pi(A)$ is an atom isomorphic to A . Further $\pi(\partial A) = \partial\pi(A)$ and $\pi(\overset{\circ}{A}) = \overset{\circ}{\pi(A)}$.*
- (b) *If $\pi(A) \neq A$, then $\pi(\overset{\circ}{A}) \cap \overset{\circ}{A} = \emptyset$.*
- (c) *If $\pi(A) \neq A$, then $\pi(A) \cap A = \partial A \cap \partial\pi(A)$.*

Proof. (a) Every automorphism permutes the set of articulations and non-trivial 2-cuts. So $\pi(\partial A)$ separates $\pi(\overset{\circ}{A})$ from the rest of the graph. It follows that $\pi(A)$ is an atom, since otherwise A would not be an atom. And π clearly preserves the boundaries and the interiors.

Chapter 3. Structural Properties of Atoms

For the rest, (b) follows from Lemma 3.5 and (c) follows from Lemma 3.6. \square

Therefore, for an automorphism π of an atom A , we require that $\pi(\partial A) = \partial\pi(A)$. If a block or proper atom $A \in \mathcal{C}$ satisfying (P2), then we can compute $\text{Aut}(A)$ according to Lemma 3.7 in polynomial time.

Projections of Atoms. Let Γ be a semiregular subgroup of $\text{Aut}(G)$, which defines a regular covering projection $p : G \rightarrow G/\Gamma$. Negami [37, p. 166] investigated possible projections of proper atoms, and we investigate this question in more details. For a proper atom or a dipole A with $\partial A = \{u, v\}$, we get one of the following three cases illustrated in Figure 3.7.

- (C1) The atom A is preserved in G/Γ , meaning $p(A) \cong A$. Notice that $p(A)$ is just a subgraph of G/Γ . For a proper atom, it can happen that $p(u)p(v)$ is adjacent, even through $uv \notin E(G)$, as in Figure 3.7.
- (C2) The interior of the atom A is preserved and the vertices u and v are identified, i.e., $p(\overset{\circ}{A}) \cong \overset{\circ}{A}$ and $p(u) = p(v)$.
- (C3) The covering projection p is a $2k$ -fold cover. There exists an involutory permutation π in Γ which exchanges u and v and preserves A . The projection $p(A)$ is a halved atom A . This can happen only when A is a halvable atom.

Lemma 3.11. *For every atom A and every semiregular subgroup Γ defining covering projection p , one of the cases (C1), (C2) and (C3) happens. Moreover, for a block atom we have exclusively the case (C1).*

Proof. For a block atom A , Lemma 3.3 implies that $p(A) \cong A$, so the case (C1) happens. It remains to deal with A being a proper atom or a dipole, and let $\partial A = \{u, v\}$. According to Lemma 3.10b every automorphism π either preserves $\overset{\circ}{A}$, or $\overset{\circ}{A}$ and $\pi(\overset{\circ}{A})$ are disjoint. If there exists a non-trivial $\pi \in \Gamma$ which preserves $\overset{\circ}{A}$, we get (C3); otherwise we get (C1) or (C2).

Let π be a non-trivial automorphism of Γ preserving $\overset{\circ}{A}$. We know $\pi(\partial A) = \partial A$ and by semiregularity, π has to exchange u and v . Then the fiber containing u and v has to be of an even size, with π being an involution reflecting k copies of A , and therefore the covering p is a $2k$ -fold cover. This proves (C3).

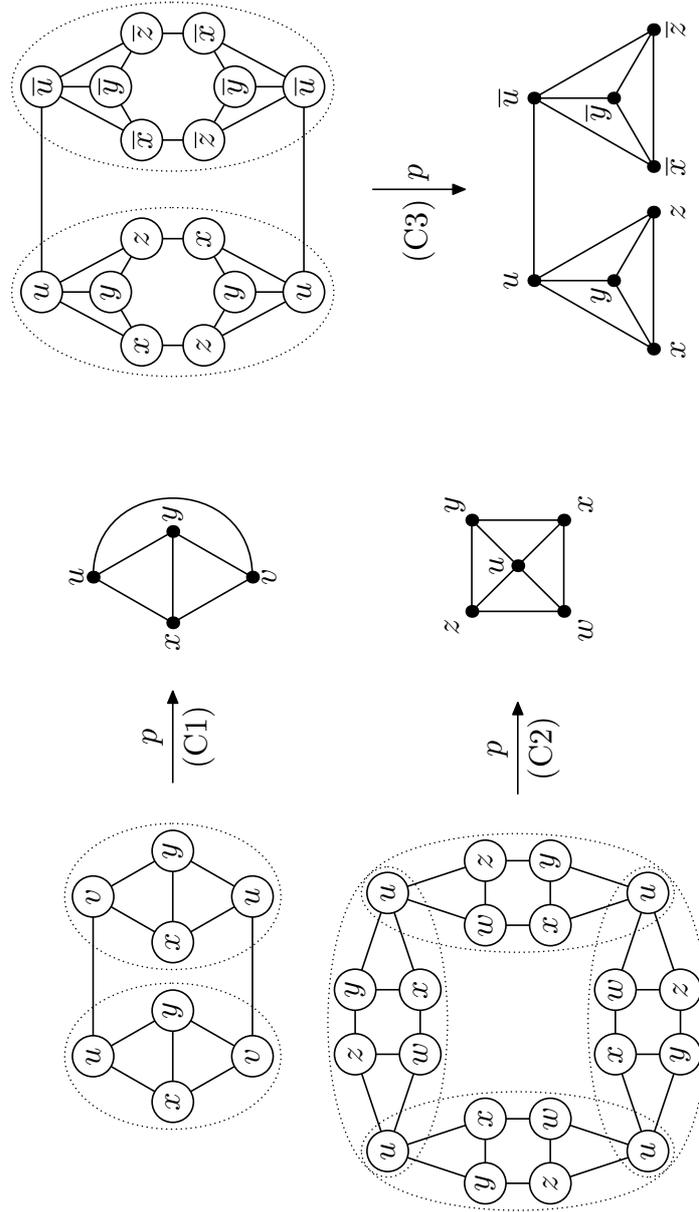


Figure 3.7: The three cases for mapping of atoms (depicted in dots). Notice that for the third graph, a projection of the type (C1) could also be applied which would give a different quotient.

Chapter 3. Structural Properties of Atoms

Suppose there is no non-trivial automorphism which preserves \mathring{A} . The only difference between (C1) and (C2) is whether u and v are contained in one fiber of p , or not. First suppose that for every non-trivial $\pi \in \Gamma$ we get $A \cap \pi(A) = \emptyset$. Then no fiber contains more than one vertex of A , and we get (C1), i.e, $A \cong p(A)$. And if there exists $\pi \in \Gamma$ such that $A \cap \pi(A) \neq \emptyset$. By Lemma 3.10c, we get $A \cap \pi(A) = \partial A \cap \partial \pi(A)$, so u and v belong to one fiber of p , which gives (C2). \square

4

Graph Reductions and Quotient Expansions

We start with a quick overview. The reduction initiates with a graph G and produces a sequence of graphs $G = G_0, G_1, \dots, G_r$. To produce G_i from G_{i-1} , we find a collection of all atoms \mathcal{A} and replace each of them by an edge of the corresponding type. We stop at step r when G_r contains no further atoms, and we call such a graph *primitive*. We call this sequence of graphs starting with G and ending with a primitive graph G_r as the *reduction series* of G .

Now suppose that $H_r = G_r/\Gamma_r$ is some quotient of G_r . To revert the reductions applied to obtain H_r , we revert the reduction series on H_r and produce an *expansion series* H_r, H_{r-1}, \dots, H_0 of H_r . We obtain semiregular subgroups $\Gamma_0, \dots, \Gamma_r$ such that $H_i = G_i/\Gamma_i$. The entire process is depicted in the following diagram:

$$\begin{array}{ccccccccccc}
 G_0 & \xrightarrow{\text{red.}} & G_1 & \xrightarrow{\text{red.}} & \dots & \xrightarrow{\text{red.}} & G_i & \xrightarrow{\text{red.}} & G_{i+1} & \xrightarrow{\text{red.}} & \dots & \xrightarrow{\text{red.}} & G_r \\
 \Gamma_0 \downarrow & & \Gamma_1 \downarrow & & & & \Gamma_i \downarrow & & \Gamma_{i+1} \downarrow & & & & \Gamma_r \downarrow \\
 H_0 & \xleftarrow{\text{exp.}} & H_1 & \xleftarrow{\text{exp.}} & \dots & \xleftarrow{\text{exp.}} & H_i & \xleftarrow{\text{exp.}} & H_{i+1} & \xleftarrow{\text{exp.}} & \dots & \xleftarrow{\text{exp.}} & H_r
 \end{array} \tag{4.1}$$

In this section, we describe structural properties of reductions and expansions. We study changes of automorphism groups done by reductions. Indeed, $\text{Aut}(G_{i+1})$ can differ from $\text{Aut}(G_i)$. But the reduction is done right and the important information of $\text{Aut}(G_i)$ is preserved in $\text{Aut}(G_{i+1})$ which is key for expansions. The issue is that expansions are unlike reductions not uniquely determined. From H_{i+1} , we can construct multiple H_i . In this section, we characterize all possible ways how H_i can be constructed from H_{i+1} .

4.1 Reducing Graphs Using Atoms

The reduction produces a series of graphs $G = G_0, \dots, G_r$. To construct G_i from G_{i-1} , we find the collection of all atoms \mathcal{A} and determine their types, using Lemma 3.9. We replace a block atom A by a pendant edge of some color based at u where $\partial A = \{u\}$. We replace each proper atom or dipole A with $\partial A = \{u, v\}$ by a new edge uv of some color and of one of the three edge types according to the type of A . According to Lemma 3.6, the replaced parts for the atoms of \mathcal{A} are pairwise disjoint, so the reduction is well defined. We stop in the step r when G_r contains no atoms. We show in Lemma 4.6 that a primitive graph is either 3-connected, a cycle, or K_2 possibly with attached single pendant edges.

To be more precise, we consider graphs with colored vertices, colored edges and with three edge types. We say that two graphs G and G' are isomorphic if there exists an isomorphism which preserves all colors and edge types, and we denote this by $G \cong G'$. We note that the results built in Section 3 transfers to colored graphs and colored atoms without any problems. Two atoms A and A' are isomorphic if there exists an isomorphism which maps ∂A to $\partial A'$. We obtain isomorphism classes for the set of all atoms \mathcal{A} such that A and A' belong to the

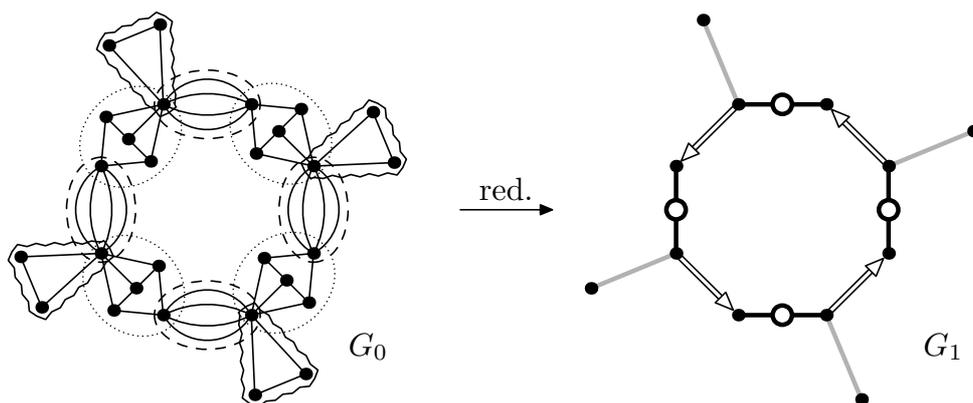


Figure 4.1: On the left, we have a graph G_0 with three isomorphism classes of atoms, each having four atoms. The dipole atoms are halvable, the block atoms are symmetric and the proper atoms are asymmetric. We reduce G_0 to G_1 which is an eight cycle with pendant leaves, with four black halvable edges, four gray undirected edges, and four white directed edges. The reduction series ends with G_1 since it is primitive.

same class if and only if $A \cong A'$. To each color class, we assign one new color not yet used in the graph. When we replace the atoms of \mathcal{A} by edges, we color the edges according to the colors assigned to the isomorphism classes.

It remains to say that for an asymmetric atom we choose an arbitrary orientation, but consistently with $\text{Aut}(G_{i-1})$ for the entire isomorphism class. For an example of the reduction, see Figure 4.1.

The symmetry type of atoms depends on the types of edges the atom contains; see Figure 4.2 for an example. Also, the figure depicts a quotient G_2/Γ_2 of G_2 , and its expansions to G_1/Γ_1 and G_0/Γ_0 . The resulting quotients G_1/Γ_1 and G_2/Γ_2 contain half-edges because Γ_1 and Γ_2 fixes some halvable edges but G_0/Γ_0 contains no half-edges. This example shows that for reductions and expansions we need to consider half-edges even when the input G and H are simple graphs.

Properties of Reduction. We want to understand how the reduction changes the automorphism group. Consider the groups $\text{Aut}(G_i)$ and $\text{Aut}(G_{i+1})$. There exists a natural homomorphism $\Phi_i : \text{Aut}(G_i) \rightarrow \text{Aut}(G_{i+1})$ which we define as follows. Let $\pi \in \text{Aut}(G_i)$. The graph G_{i+1} is constructed from G_i by replacing interiors of all atoms by colored edges. For the common vertices and edges of G_i and G_{i+1} , we define the image in $\Phi_i(\pi)$ exactly as in π . If A is an atom of G_i , then according to Lemma 3.10a, $\pi(A)$ is an atom isomorphic to A . In G_{i+1} , we replace the interiors of both A and $\pi(A)$ by the edges e_A and $e_{\pi(A)}$ of the same type and color. Therefore, we define $\Phi_i(\pi)(e_A) = e_{\pi(A)}$.

More precisely for purpose of Section 4.2, we define Φ_i on the half edges. Let $e_A = uv$ and let h_u and h_v be the half-edges composing e_A , and similarly let $h_{\pi(u)}$ and $h_{\pi(v)}$ be the half-edges composing $e_{\pi(A)}$. Then we define $\Phi_i(\pi)(h_u) = h_{\pi(u)}$ and $\Phi_i(\pi)(h_v) = h_{\pi(v)}$.

Proposition 4.1. *The mapping Φ_i satisfies the following:*

- (a) *The mapping Φ_i is a group homomorphism.*
- (b) *The mapping Φ_i is surjective.*
- (c) *Moreover, $\text{Aut}(G_{i+1}) = \Phi_i(\text{Aut}(G_i))$ monomorphically embeds into $\text{Aut}(G_i)$.*
- (d) *For a semiregular subgroup Γ of $\text{Aut}(G_i)$, the mapping $\Phi_i|_{\Gamma}$ is an isomorphism. Moreover, the subgroup $\Phi_i(\Gamma)$ remains semiregular.*

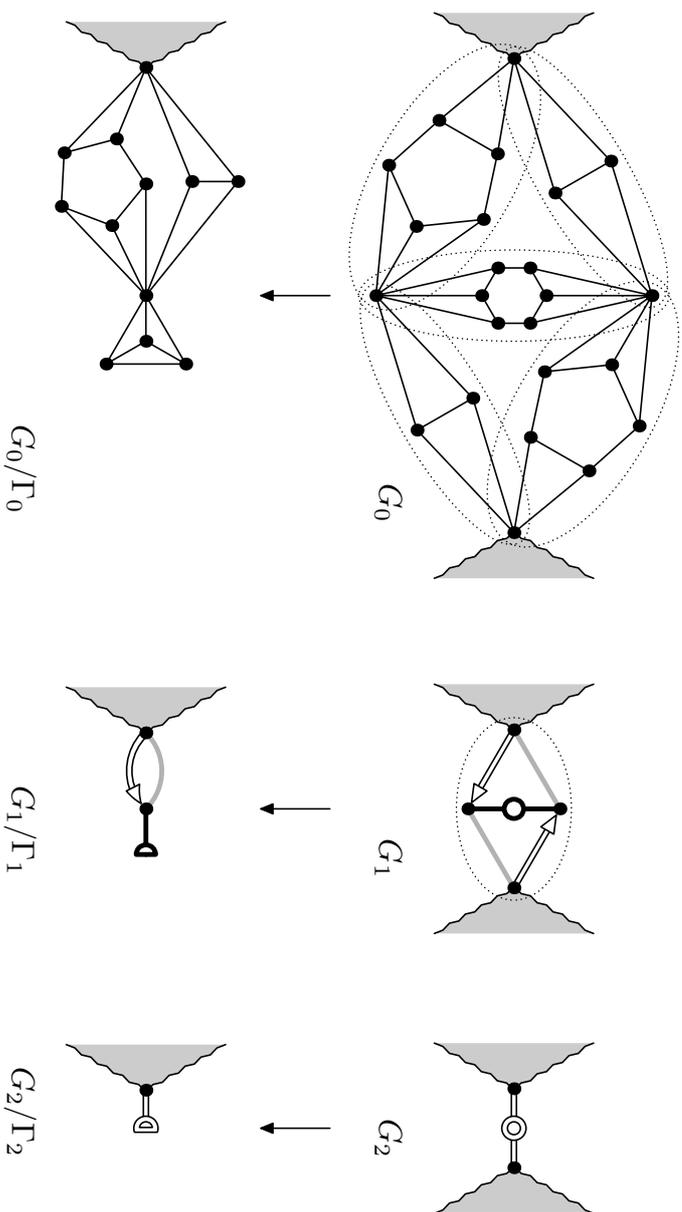


Figure 4.2: We reduce a part of a graph in two steps. In the first step, we replace five atoms by five edges of different types. As the result we obtain one halvable atom which we further reduce to one halvable edge. Notice that without considering edge types, the resulting atom would be just symmetric. In the bottom we show the corresponding quotient graphs when Γ_i are generated by semiregular involutory automorphism π from the definition of the halvable atom.

4.1. Reducing Graphs Using Atoms

Proof. (a) It is easy to see that each $\Phi_i(\pi) \in \text{Aut}(G_{i+1})$. The well-known Homomorphism Theorem states that Φ_i is a homomorphism if and only if the kernel $\text{Ker}(\Phi_i)$, i.e., the set of all π such that $\Phi_i(\pi) = \text{id}$, is a normal subgroup of $\text{Aut}(G_i)$. It is easy to see that the kernel $\text{Ker}(\Phi_i)$ has the following structure. If $\pi \in \text{Ker}(\Phi_i)$, it fixes everything except for the interiors of the atoms. Further, $\pi(\overset{\circ}{A}) = \overset{\circ}{\pi}(A)$, so π can non-trivially act only inside the interiors of the atoms.

Let $\sigma \in \text{Aut}(G_i)$ and $\pi \in \text{Ker}(\Phi_i)$. We need to show that $\sigma\pi\sigma^{-1} \in \text{Ker}(\Phi_i)$. Let A be an atom. Then $\sigma(A) = A'$ is an isomorphic atom. The composition clearly permutes the interior $\overset{\circ}{A}$. Moreover, the part of the graph outside of interiors is fixed by the composition. Hence it belongs to $\text{Ker}(\Phi_i)$ and by the Homomorphism Theorem Φ_i is a homomorphism.

(b) Let $\pi' \in \text{Aut}(G_{i+1})$, we want to extend π' to $\pi \in \text{Aut}(G_i)$ such that $\Phi_i(\pi) = \pi'$. We just need to describe this extension on a single edge $e = uv$. If e is an original edge of G , there is nothing to extend. Suppose that e was created in G_{i+1} from an atom A in G_i . Then $e' = \pi'(e)$ is an edge of the same color and the same type as e , and therefore e' is constructed from an isomorphic atom A' of the same symmetry type. The automorphism π' prescribes the action on the boundary ∂A . We need to show that it is possible to define an action on $\overset{\circ}{A}$ consistently.

If A is a block atom, then the both edges e and e' are pendant, attached by articulations u and u' . We just define π using an isomorphism from A to A' which takes u to u' . It remains to deal with proper atoms and dipoles.

First suppose that A is an asymmetric atom. Then by definition the orientation of e and e' is consistent with respect to π' . Since $\overset{\circ}{A} \cong \overset{\circ}{A}'$, we define π on $\overset{\circ}{A}$ according to one such isomorphism.

Secondly suppose that A is symmetric or halvable. Let σ be an isomorphism of A and A' . Either σ maps ∂A exactly as π' , and then we can use σ for defining π . Or we compose σ with the automorphism of A exchanging the two vertices of ∂A . (We know that such an automorphism exists since A is not antisymmetric.)

(c) Let $(A_1, A_2, \dots, A_\ell)$ be an orbit of π on edges representing atoms A_1, \dots, A_ℓ . We construct the extension as above, choosing any isomorphism from A_i to A_{i+1} , where $i = 1, \dots, \ell - 1$, and properly

the isomorphism A_ℓ to A_1 . Here, properly means that composition of all these isomorphisms is the identity on A_1 . Repeating this procedure for every orbit of π , we determine an extension $\tilde{\pi}$ of π defining a monomorphic embedding $\pi \mapsto \tilde{\pi}$.

(d) We note that for any subgroup Γ , the restricted mapping $\Phi_i|_\Gamma$ is a group homomorphism with $\text{Ker}(\Phi_i|_\Gamma) = \text{Ker}(\Phi_i) \cap \Gamma$. If Γ is semiregular, then $\text{Ker}(\Phi_i) \cap \Gamma$ is trivial. The reason is that G_i contains at least one atom A , and the boundary ∂A is fixed by $\text{Ker}(\Phi_i)$. Hence $\Phi_i|_\Gamma$ is an isomorphism.

For the semiregularity of $\Phi_i(\Gamma)$, let π' be an automorphism of G_{i+1} . Since $\Phi_i|_\Gamma$ is an isomorphism, there exists the unique $\pi \in \Gamma$ such that $\Phi_i(\pi) = \pi'$. If π' fixes a vertex u , then π fixes u as well, so it is the identity, and $\pi' = \Phi_i(\text{id}) = \text{id}$. And if π' only fixes an edge $e = uv$, then π' exchanges u and v . Since π does not fix e , then there is an atom A replaced by e in G_{i+1} . Then $\pi|_A$ is an involutory semiregular automorphism exchanging u and v , so A is halvable. But then e is a halvable edge, and thus π' can fix it. \square

The above statement is an example of a phenomenon known in permutation group theory. Interiors of atoms behave as *blocks of imprimitivity* in the action of $\text{Aut}(G_i)$. It is well-known that the kernel of the action on the imprimitivity blocks is a normal subgroup of $\text{Aut}(G_i)$. For the example of Figure 4.2, we get $\text{Aut}(G_1) = \mathbb{C}_3$ and $\text{Ker}(\Phi_1) = \mathbb{C}_2^3 \times \mathbb{S}_3^3$. As a simple corollary, we get:

Corollary 4.2. *We get*

$$\text{Aut}(G_r) = \left(\cdots \left((\text{Aut}(G_0)/\text{Ker}(\Phi_1))/\text{Ker}(\Phi_2) \right) \cdots / \text{Ker}(\Phi_r) \right).$$

Proof. We already proved that $\text{Aut}(G_{i+1}) = \text{Aut}(G_i)/\text{Ker}(\Phi_i)$. \square

Actually, one can prove much more, that $\text{Aut}(G_i) = \text{Aut}(G_{i+1}) \rtimes \text{Ker}(\Phi_i)$. First, we describe the structure of $\text{Ker}(\Phi_i)$.

Lemma 4.3. *The group $\text{Ker}(\Phi_i)$ is the direct product $\prod_{A \in \mathcal{A}} \text{Fix}(A)$ where $\text{Fix}(A)$ is the point-wise stabilizer of $G_i \setminus \overset{\circ}{A}$ in $\text{Aut}(G_i)$.*

Proof. According to Lemma 3.5, the interiors of the atoms are pairwise disjoint, so $\text{Ker}(\Phi_i)$ acts independently on each interior. Thus we get

4.1. Reducing Graphs Using Atoms

$\text{Ker}(\Phi_i)$ as the direct product of actions on each interior $\overset{\circ}{A}$ which is precisely $\text{Fix}(A)$. \square

Alternatively, $\text{Fix}(A)$ is isomorphic to the point-wise stabilizer of ∂A in $\text{Aut}(A)$. Let A_1, \dots, A_s be pairwise non-isomorphic atoms in G_i , each appearing with the multiplicity m_i . According to Lemma 4.3, we get $\text{Ker}(\Phi_i) \cong \text{Fix}(A_1)^{m_1} \times \dots \times \text{Fix}(A_s)^{m_s}$.

Proposition 4.4. *We get*

$$\text{Aut}(G_i) \cong \text{Aut}(G_{i+1}) \rtimes \text{Ker}(\Phi_i).$$

Proof. According to Proposition 4.1c, we know that $\text{Ker}(\Phi_i) \triangleleft \text{Aut}(G_i)$ has a complement isomorphic to $\text{Aut}(G_{i+1})$. Actually, this already proves that $\text{Aut}(G_i)$ has the structure of the semidirect product. We give more details into its structure.

Each element of $\text{Aut}(G_i)$ can be written as a pair (π, σ) where $\pi \in \text{Aut}(G_i)$ and $\sigma \in \text{Ker}(\Phi_i)$. We first apply π and permute G_i , mapping interiors of the atoms as blocks. Then σ permutes the interiors of the atoms, preserving the remainder of G_i .

It remains to understand how composition of two automorphisms (π, σ) and $(\hat{\pi}, \hat{\sigma})$ works. We get this as a composition of four automorphisms $\hat{\sigma} \circ \hat{\pi} \circ \sigma \circ \pi$, which we want to write as a pair (τ, ρ) . Therefore, we need to swap $\hat{\pi}$ with σ . This clearly preserves $\hat{\pi}$, since the action $\hat{\sigma}$ on the interiors does not influence it; so we get $\tau = \hat{\pi} \circ \pi$.

But σ is changed by this swapping. According to Lemma 4.3, we get $\sigma = (\sigma_1, \dots, \sigma_s)$ where each $\sigma_i \in \text{Fix}(A_i)^{m_i}$. Since π preserves the isorphism classes of atoms, it acts on each σ_i independently and permutes the isomorphic copies of A_i . Suppose that A and A' are two isomorphic copies of A_i and $\pi(A) = A'$. Then the action of σ_i on the interior of A corresponds after the swapping to the same action on the interior of $A' = \pi(A)$. This can be described using the semidirect product, since each π defines an automorphism of $\text{Ker}(\Phi_i)$ which permutes the coordinates of each $\text{Fix}(A_i)^{m_i}$. \square

We note that in a similar manner, Babai [4, 5] characterized automorphism groups of planar graphs.

Lemma 4.5. *Let G admit a non-trivial semiregular automorphism π . Then each G_i has a central block which is obtained from the central block of G_{i-1} by replacing its atoms by colored edges.*

Proof. By Proposition 4.1d existence of a semiregular automorphism is preserved during the reduction. Thus by Lemma 3.2, each G_i has a central block. Since we replace only proper atoms and dipoles in the central block, it remains as a block after reduction. We argue by induction that it remains central as well.

Let B be the central block of G_i and let B' be this block in G_{i+1} . Consider the subtree R'_u of the block tree T' of G_{i+1} attached to B in u containing the longest path in T' from B . This subtree corresponds to R_u in G_i . (See Section 3.1 for the definition of R_u .) Let π be a non-trivial semiregular automorphism in G_i . Then $\pi(u) = v$, and by Lemma 3.3 we have $R_v \cong R_u$. Then R'_v corresponds in G_{i+1} to R_v after reduction and $R'_u \cong R'_v$. Therefore B' is the central block of G_{i+1} . \square

Primitive Graphs. Recall that a graph is called primitive if it contains no atoms. If G has a non-trivial semiregular automorphism, then according to Lemma 4.5 the central block is preserved in the primitive graph G_r . We shall assume in the following that every primitive graph has a central block.

Lemma 4.6. *Let G be a graph with a central block. Then the graph G is primitive if and only if it is isomorphic to a 3-connected graph, to a cycle C_n for $n \geq 2$, or to K_2 , or can be obtained from these graphs by taking $U \subseteq V(G)$ such that $|U| \geq 2$ and attaching a single pendant edge to each vertex of U .*

Proof. The primitive graphs are depicted in Figure 4.3 and clearly such graphs are primitive. For the other implication, the graph G contains a central block. All blocks attached to it have to be single pendant edges,

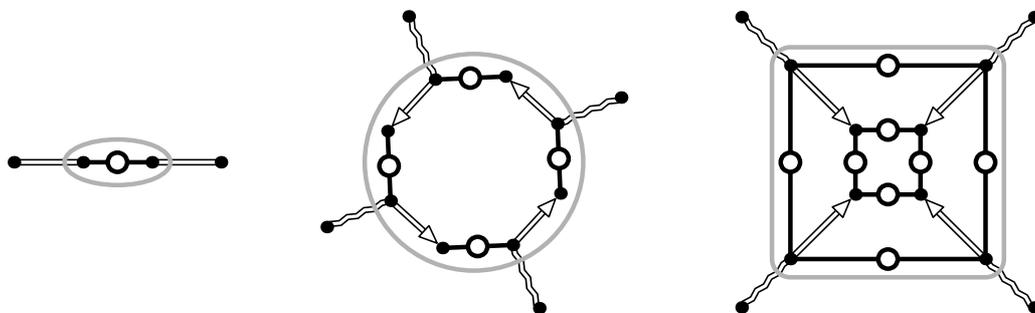


Figure 4.3: A primitive graph with a central block is either K_2 , C_n , or a 3-connected graph, in all three cases with possible single pendant edges attached to it.

4.2. Quotients and Their Expansion

otherwise G would contain a block atom. By removal of all pendant edges, we get the 2-connected graph B consisting of only the central block. We argue that B is isomorphic to one of the graphs above.

Now, let u be a vertex of the minimum degree of B . If $\deg(u) = 1$, the graph B has to be K_2 , otherwise it would not be 2-connected. If $\deg(u) = 2$, then either the graph B is a cycle C_n , or u is an inner vertex of a path connecting two vertices x and y of degree at least three such that all inner vertices are of degree two. But then this path is an atom. And if $\deg(u) \geq 3$, then every 2-cut is non-trivial, and since B contains no atoms, it has to be 3-connected. \square

We note that if existence of a central block is not required, and we define atoms with respect to the central articulation then in addition the primitive graph can be K_1 .

4.2 Quotients and Their Expansion

Let G_0, \dots, G_r be the reduction series of G and let Γ_0 be a semiregular subgroup of $\text{Aut}(G_0)$. By repeated application of Proposition 4.1d, we get the uniquely determined semiregular subgroups $\Gamma_1, \dots, \Gamma_r$ of $\text{Aut}(G_1), \dots, \text{Aut}(G_r)$, each isomorphic to Γ_0 . Let $H_i = G_i/\Gamma_i$ be the quotients where we preserve colors of edges in the quotients, and let p_i be the corresponding covering projection from G_i to H_i . Recall that H_i can contain edges, loops and half-edges; depending on the action of Γ_i on the half-edges corresponding to the edges of G_i .

Quotients Reductions. Consider $H_i = G_i/\Gamma_i$ and $H_{i+1} = G_{i+1}/\Gamma_{i+1}$. We investigate relations between these quotients. Let A be an atom of G_i represented by a colored edge e in G_{i+1} . According to Lemma 3.11, we have three possible cases (C1), (C2) and (C3) for the projection $p_i(A)$. It is easy to see that Φ_i is defined exactly in the way that $p_{i+1}(e)$ corresponds to an edge in the case (C1), to a loop in the case (C2) and to a half-edge in the case (C3). See Figure 4.4 for examples. In other words, we get the following commuting diagram:

$$\begin{array}{ccc}
 G_i & \xrightarrow{\text{red.}} & G_{i+1} \\
 \Gamma_i \downarrow & & \downarrow \Gamma_{i+1} \\
 H_i & \xrightarrow{\text{red.}} & H_{i+1}
 \end{array} \tag{4.2}$$

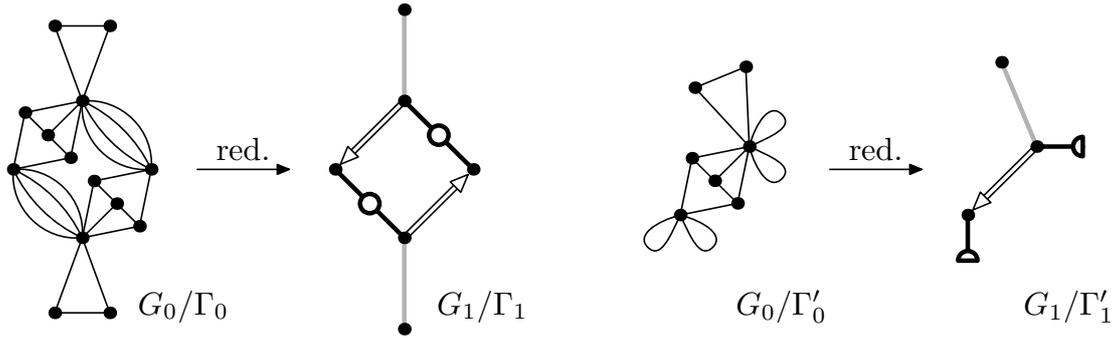


Figure 4.4: Example of two quotients of the graph G_0 from Figure 4.1 with the corresponding quotients of the reduced graph G_1 . Here $\Gamma_1 = \Phi_1(\Gamma_0)$ and $\Gamma'_1 = \Phi_1(\Gamma'_0)$.

So we can construct the graph H_{i+1} from H_i by replacing the projections of atoms in H_i by the corresponding projections of the edges replacing the atoms. We get the following.

Lemma 4.7. *Every semiregular subgroup Γ_i of $\text{Aut}(G_i)$ corresponds to a unique semiregular subgroup Γ_{i+1} of $\text{Aut}(G_{i+1})$.*

Overview of Quotients Expansions. Our goal is to reverse the horizontal edges in Diagram (4.2), i.e, to understand:

$$\begin{array}{ccc}
 G_i & \xleftarrow{\text{exp.}} & G_{i+1} \\
 \Gamma_i \downarrow & & \downarrow \Gamma_{i+1} \\
 H_i & \xleftarrow{\text{exp.}} & H_{i+1}
 \end{array} \tag{4.3}$$

Now we investigate the opposite relations. There are two fundamental questions we address in this section in full details:

- *Question 1.* Given a group Γ_{i+1} , how many different semiregular groups Γ_i do we have such that $\Phi_i(\Gamma_i) = \Gamma_{i+1}$? Notice that all these groups Γ_i are isomorphic to Γ_{i+1} as abstract groups, but they correspond to different actions on G_i .
- *Question 2.* Let Γ_i and Γ'_i be two groups extending Γ_{i+1} . Under which conditions are the quotients $H_i = G_i/\Gamma_i$ and $H'_i = G_i/\Gamma'_i$ different graphs?

Extensions of Group Actions. We first deal with Question 1. Let Γ_i and Γ_{i+1} be the semiregular groups such that $\Phi_i(\Gamma_i) = \Gamma_{i+1}$. Then we call Γ_{i+1} a *reduction* of Γ_i , and Γ_i an *extension* of Γ_{i+1} .

4.2. Quotients and Their Expansion

Lemma 4.8. *For every semiregular group Γ_{i+1} , there exists an extension Γ_i .*

Proof. First notice that Γ_{i+1} determines the action of Γ_i everywhere on G_i except for the interiors of the atoms, so we just need to define it there. Let $e = uv$ be one edge of G_{i+1} replacing an atom A in G_i . First, we assume that A is not a block atom. Let $|\Gamma_{i+1}| = k$. We distinguish two cases. Either the orbit $[e]$ contains exactly k edges, or it contains $\frac{k}{2}$ edges. See Figure 4.5 for an overview.

Case 1: The orbit $[e]$ contains exactly k edges. Let e_1, \dots, e_k be the orbit $[e]$ and let $u_i = \pi'(u)$ and $v_i = \pi'(v)$ for the unique π' mapping e to e_i . (We know that π' is unique because Γ_{i+1} is semiregular.) Let A_1, \dots, A_k be the atoms corresponding to e_1, \dots, e_k . The edges e_1, \dots, e_k have the same color and type, and thus the atoms A_i are pairwise isomorphic and of the same type.

We define the action of Γ_i on the interiors of A_1, \dots, A_k as follows. Let $\sigma_{1,i}$ denote any isomorphism from A_1 to A_i such that $\sigma_{1,i}(u_1) = u_i$ and $\sigma_{1,i}(v_1) = v_i$, with $\sigma_{1,1}$ being the identity on A_1 . Such isomorphism exists trivially for symmetric and halvable atoms, and they also exists for asymmetric atoms since the action of Γ_{i+1} preserves the orientation of e_1, \dots, e_k . Then we define $\sigma_{i,j} = \sigma_{1,j}\sigma_{1,i}^{-1}$. Let $\pi' \in \Gamma_{i+1}$ and we define the extension π as follows. If π' maps e_i to e_j , we set $\pi|_{A_i} = \sigma_{i,j}$.

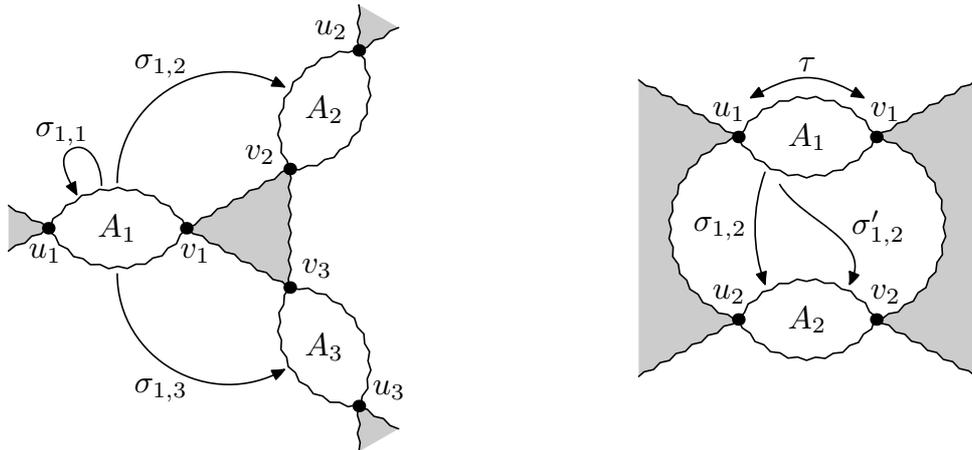


Figure 4.5: Case 1 is depicted on the left for three edges corresponding to isomorphic atoms A_1 , A_2 and A_3 . The depicted isomorphisms are used to extend Γ_{i+1} on interiors of these atoms. Case 2 is on the right, with additional semiregular involution τ from the definition of halvable atoms which transposes u_1 and v_1 .

Case 2: The orbit $[e]$ contains exactly $\ell = \frac{k}{2}$ edges. Then we have k half-edges in one orbit, so in H_i we get one half-edge. Let e_1, \dots, e_ℓ be the edges of $[e]$. They have to be halvable, and consequently the corresponding atoms A_1, \dots, A_ℓ are halvable. Let u_i be an arbitrary endpoint of e_i and let v_i be the second endpoint of e_i . Let τ be any involutory semiregular automorphism of A_1 which maps u_1 to v_1 ; we know that such τ exists since A_1 is a halvable atom.

Similarly as above, we set $\sigma_{1,i}$ to be any isomorphism mapping A_1 to A_i such that $\sigma_{1,i}(u_1) = u_i$, and we put $\sigma_{1,1} = \text{id}$. Moreover, we put $\sigma'_{1,1} = \tau$ and $\sigma'_{1,i} = \sigma_{1,i}\tau$. Then we put $\sigma_{i,j} = \sigma_{1,j}\sigma_{1,i}^{-1}$, and $\sigma'_{i,j} = \sigma'_{1,j}\sigma_{1,i}^{-1}$. Let $\pi' \in \Gamma_{i+1}$ and $\pi'(e_i) = e_j$. To define the extension π , we set $\pi|_{A_i}$ equal $\sigma_{i,j}$ if $\pi'(u_i) = u_j$, and $\sigma'_{i,j}$ if $\pi'(u_i) = v_j$.

We deal with block atoms in a similar manner as in Case 1, except the orbit $[u]$ consists of articulations, and the orbit $[v]$ consists of leaves. It is easy to observe that by semiregularity of Γ_{i+1} the constructed group Γ_i acts semiregularly on G_i , as well. \square

Corollary 4.9. *The construction in the above proof gives all possible extensions of Γ_{i+1} .*

Proof. We get all possible choices for Γ_i in Case 1 by different choices of $\sigma_{1,i}$, and in Case 2 by different choices of $\sigma_{1,i}$ and τ . \square

Quotients of Atoms. To answer Question 2, we first need to understand possible quotients of an atom A . In Section 3.4, we stated that that for each regular covering projection $p : G_i \rightarrow H_i$, the projection $p(A)$ satisfies one of the three cases (C1), (C2) and (C3). Figure 4.6 shows how $p(A)$ can look in H_i depending on which of the three cases happens. If A is a block atom, it is always projected as in the case (C1).

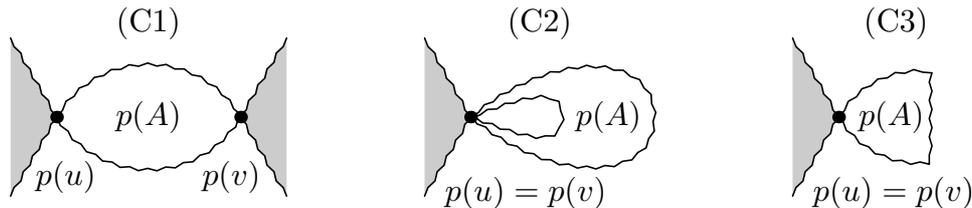


Figure 4.6: How can $p(A)$ look in G_i/Γ_i , depending on the cases (C1), (C2) and (C3).

4.2. Quotients and Their Expansion

So we get three types of quotients $p(A)$ of A . For (C1), we call this quotient an *edge-quotient*, for (C2) a *loop-quotient* and for (C3) a *half-quotient*. The reason lying behind these names is that $p(A)$ is in H_{i+1} represented by an edge, a loop or a half-edge respectively. The following lemma allows to say “the” edge- and “the” loop-quotient of an atom.

Lemma 4.10. *For every atom A , there is the unique edge-quotient and the unique loop-quotient up to isomorphism.*

Proof. For the cases (C1) and (C2), we have $\dot{A} \cong \dot{p}(A)$, so the quotients are unique. □

For half-quotients uniqueness does not hold. First, an atom A has to be halvable to admit a half-quotient. Then each half-quotient is determined by an involutory automorphism, and we denote τ its restriction to A ; recall (C3). There is a one-to-many relation between non-isomorphic half-quotients and automorphisms τ , i.e., several different automorphisms τ may give the same half-quotient.

For a proper atom, we can bound the number of non-isomorphic half-quotients by the number of different semiregular involutions of 3-connected graphs.

Lemma 4.11. *Let A be a proper atom of the class \mathcal{C} satisfying (P2). Then there are polynomially many non-isomorphic half-quotients of A .*

Proof. The graph A^+ is essentially 3-connected graph and belongs to \mathcal{C} . According to (C2), the number of different semiregular subgroups of order two is polynomial in the size of A^+ . Each half-quotient is defined by one of these semiregular involutions which fix the edge uv and transpose u and v . □

For dipoles, we get the following result valid for general graphs:

Lemma 4.12. *Let A be a dipole. Then the number of pairwise non-isomorphic half-quotients is at most $2^{\lfloor \frac{e(A)}{2} \rfloor}$ and this bound is achieved.*

Proof. Figure 4.7 shows a construction of dipoles achieving the upper bound. It remains to argue correctness of the upper bound.



Figure 4.7: An example of a dipole with a pair of white halvable edges and a pair of black halvable edges (corresponding to two isomorphism classes of halvable atoms). There exist four pairwise non-isomorphic half-quotients. This example can easily be generalized to exponentially many pairwise non-isomorphic quotients by introducing more pairs of halvable edges of additional colors.

First, we derive the structure of all involutory semiregular automorphisms τ acting on $\overset{\circ}{A}$. We have no freedom concerning the non-halvable edges of A : The undirected edges of each color class has to be paired by τ together. Further, each directed edge has to be paired with a directed edges of the opposite direction and the same color. It remains to describe possible action of τ on the remaining at most $e(A)$ halvable edges of A . These edges belong to c color classes having m_1, \dots, m_c edges. Each automorphism τ has to preserve the color classes, so it acts independently on each class.

We concentrate only for one color class having m_i edges. We bound the number $f(m_i)$ of pairwise non-isomorphic quotients of this class. Then we get the upper bound

$$\prod_{1 \leq i \leq c} f(m_i) \tag{4.4}$$

for the number of non-isomorphic half-quotients of A .

An edge e fixed in τ is mapped into a half-edge of the given color in the half-quotient of A . And if τ maps e to $e' \neq e$, then we get a loop in the half-quotient. The resulting half-quotient only depends on the number of fixed edges and fixed two-cycles in the considered color class. We can construct at most $f(m_i) = \lfloor \frac{m_i}{2} \rfloor + 1$ pairwise non-isomorphic quotients, since we may have zero to $\lfloor \frac{m_i}{2} \rfloor$ loops with the complementing number of half-edges.

The bound (4.4) is maximized when each class contains exactly two edges. (Except for one class containing three or one edge if m is odd.) \square

This bound plays the key role for the complexity of our meta-

4.2. Quotients and Their Expansion

algorithm of Section 5; in one subroutine, we iterate over all half-quotients of a dipole. Also the structure of all possible quotients is important.

Quotient Expansion. When we know all quotients of atoms, we can construct from given H_{i+1} all quotients H_i as follows. We say that two quotients H_i and H'_i extending H_{i+1} are different if there exists no isomorphism of H_i and H'_i which fixes the vertices and edges common with H_{i+1} . (But H_i and H'_i still might be isomorphic.)

Proposition 4.13. *Every quotient H_i of G_i can be constructed from some quotient H_{i+1} of G_{i+1} by replacing each edge, loop and half-edge corresponding to an atom of G_i by an edge-, loop-, or half-quotient respectively. Moreover, for different choices of H_{i+1} and of half-quotients we get different graphs H_i .*

Proof. Let $H_{i+1} = G_{i+1}/\Gamma_{i+1}$ and let H_i be constructed in the above way. We first argue that H_i is a quotient of G_i , i.e., it is equal to G_i/Γ_i for some Γ_i extending Γ_{i+1} . To see this, it is enough to construct Γ_i in the way described in the proof of Lemma 4.8. We choose $\sigma_{1,i}$ arbitrarily, and the involutory permutations τ are prescribed by chosen half-quotients replacing half-edges. It is easy to see that the resulting graph is the constructed H_i . We note that only the choices of τ matter, for arbitrary choices of $\sigma_{1,i}$ we get the same quotients.

On the other hand, if H_i is a quotient, it replaces the edges, loops and half-edges of H_{i+1} by some quotients, so we can generate H_i in this way. The reason is that according to Corollary 4.9, we can generate all Γ_i extending Γ_{i+1} by some choices $\sigma_{1,i}$ and τ .

For the last statement, according to Lemma 4.10, the edge and loop-quotients are uniquely determined, so we are only free in choosing different half-quotients. For different choices of half-quotients, we get different graphs H_i . □

For instance suppose that H_{i+1} contains a half-edge corresponding to the dipole from Figure 4.7. Then in H_i we can replace this half-edge by one of the four possible half-quotients of this dipole.

Corollary 4.14. *If H_{i+1} contains no half-edge, then H_i is uniquely determined. So for odd order of Γ_r , the quotient H_r uniquely determines H_0 .*

Chapter 4. Graph Reductions and Quotient Expansions

Proof. Implied by Proposition 4.13 and Lemma 4.10 which states that edge- and loop-quotients are uniquely determined. If the order of Γ_r is odd, no half-edges are constructed. \square

The Block Structure of Quotients. The following properties are key for identifying quotients of atoms in the input graph H . The approach used in the meta-algorithm is to find a way how to expand H_r by repeated application of Proposition 4.13 to H_0 which is isomorphic to the input H .

A block atom A of G_i is always projected by (C1), and so it corresponds to a block atom of H_i . Suppose that A is a proper atom or a dipole, and let $\partial A = \{u, v\}$. For (C1) we get $p(u) \neq p(v)$, and for (C2) and (C3) we get $p(u) = p(v)$. For (C1), $p(A)$ is isomorphic to an atom in H_i . For (C2) and (C3) is $p(u)$ an articulation of H_i , and $p(A)$ corresponds to a pendant star, or a pendant block with possible attached single pendant edges.

Lemma 4.15. *The block structure of H_{i+1} is preserved in H_i with possible some new pendant blocks attached.*

Proof. Edges inside blocks are replaced using (C1) by edge-quotients of block atoms, proper atoms and dipoles which preserves 2-connectivity. The new pendant blocks in H_i are created by replacing pendant edges with the block atoms, loops by loop-quotients, and half-edges by half-quotients. \square

5

Meta-algorithm

In this section, we establish the fixed parameter tractable algorithm of Theorem 1.1. We show that for a class \mathcal{C} satisfying (P0) to (P3) we can solve $\text{REGULARCOVER}(G, H)$ in time $\mathcal{O}^*(2^{e(H)/2})$. We use the property (P3) for essentially 3-connected graphs with colored pendant edges which code colors and lists of colors.

Let $k = |G|/|H|$, and we assume that $k \geq 2$. (If k is not an integer, then clearly G does not cover H . If $k = 1$, then we can test it using the algorithm for graph isomorphism given by (P1) whether $G \cong H$.) The algorithm proceeds in the following major steps:

1. We construct the reduction series for $G = G_0, \dots, G_r$ terminating with the unique primitive graph G_r . Throughout the reduction the central block is preserved, otherwise according to Lemma 4.5 there exists no semiregular automorphism of G and we output “no”. According (P0), the reduction preserves the class \mathcal{C} , and also every atom belongs to \mathcal{C} .
2. Using (P2), we compute $\text{Aut}(G_r)$ and construct a list of all subgroups Γ_r of the order k acting semiregularly on G_r . The number of subgroups in the list is polynomial by (P2).
3. For each Γ_r in the list, we compute $H_r = G_r/\Gamma_r$. We say that a graph H_r is *expandable* if there exists a sequence of extensions repeatedly applying Proposition 4.13 which constructs H_0 isomorphic to H . We test the expandability of H_r using dynamic programming while using (P1) and (P3).

It remains to explain details of the third step, and prove the correctness of the algorithm.

5.1 Testing Expandability Using Dynamic Programming

Catalog of Atoms. During the reduction phase of the algorithm, we construct the following *catalog of atoms* forming a database of all discovered atoms and their quotients. We are not very concerned with a specific implementation of the algorithm, so this catalog is mainly used to simplify description. For each isomorphism class of atoms represented by an atom A , we store the following information in the catalog:

- the atom A ,
- the corresponding colored edge of a given type representing the atom in the reduction,
- the unique edge- and loop-quotients of A .

For dipoles, according to Lemma 4.12 we can have exponentially many non-isomorphic half-quotients, and so we work with their half-quotients implicitly in the dynamic programming.

If A is not a dipole, we compute a list of all its pairwise non-isomorphic half-quotients, and store them in the catalog in the following way. A half-quotient Q might not be 3-connected, and so we apply a reduction series on Q , and add all atoms discovered by the reduction to the catalog. (We do not compute their half-quotients. They are never realized unless these atoms are directly found in G as well.) When the reduction series finishes, this half-quotient is reduced to a primitive graph. We note that ∂Q , being a single vertex of the half-quotient, behaves like the central block in the definition of atoms, i.e., it is never reduced.

Further, if a halvable dipole consists of exactly two edges of the same color, we compute its half-quotient consisting of just the single loop attached, and we add this quotient to the catalog. The reason is that that this quotient behaves exactly as a loop-quotient of some proper atom.

Lemma 5.1. *The catalog contains polynomially many quotients and atoms.*

5.1. Testing Expandability Using Dynamic Programming

Proof. First we deal with the number of atoms in G_0, \dots, G_r . Notice that by replacing an interior of an atom, the total number of vertices and edges is decreased; the interiors of atoms in each G_i contain at least two vertices and edges in total and are pairwise disjoint (implied by Lemma 3.5). Thus we have $\mathcal{O}(n + m)$ atoms in G_0, \dots, G_r .

For the number of quotients, let A be a block or a proper atom. Each half-quotient of A is created by some semiregular action of an involution on A . According to (P2), there are polynomially many half-quotients. For each half-quotient, we can have at most linearly many atoms in its reduction series. And by Lemma 4.10 we have the unique edge- and loop-quotient. So the total number of atoms and their quotients is polynomial. \square

Throughout the algorithm, we repeatedly ask whether some atom or some of its quotients is contained in the catalog. Each such query can be answered in polynomial time.

Preimages of a Pendant Block. We now illustrate the fundamental difficulty in testing whether H_r is expandable to H , for simplicity we do it on pendant blocks. Suppose that H has a pendant block as in Figure 5.1. Then there is no way to know whether this block corresponds in G to an edge-quotient of a block atom, or to a loop-quotient of a proper atom, or to a half-quotient of another proper atom. It can easily happen that the catalog offers all three options. So without exploiting some additional information from H , there is no way to know what is the preimage of this pendant block.

In our approach, we do not decide everything in one stage, instead we just remember a list of possibilities. The dynamic programming deals with these lists and computes further lists for larger parts of H .

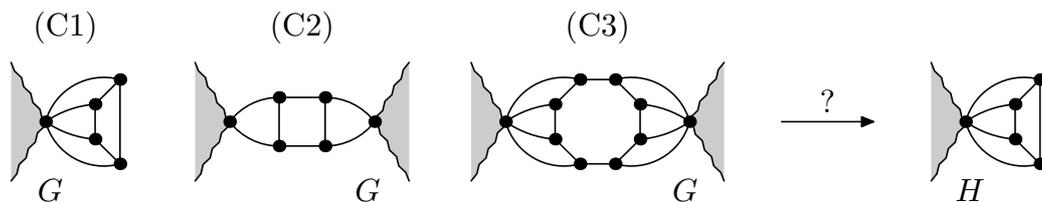


Figure 5.1: For a pendant block of H , there are three possible preimages in G . It could be a block atom mapped by (C1), or a proper atom mapped by (C2), or another proper atom mapped by (C3) (where the half-quotient is created by 180° rotation τ).

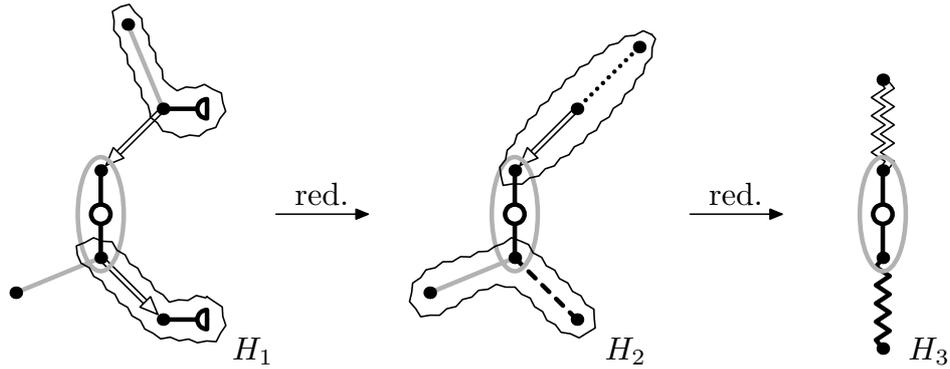


Figure 5.2: The graph H_1 is one quotient of G_1 from Figure 4.1. We further reduce it to H_3 with respect to the core block depicted in gray. Notice that H_1 and H_2 only contain block atoms.

Atoms in Quotients. We define atoms in the quotient graphs similarly as in Section 3 with only one difference. We choose one arbitrary block/articulation called the *core* in H_r ; for instance, we can choose the central block/articulation. The core plays the role of the central block in the definition of parts and atoms. Also, in the definition we consider half-edges and loops as pendant edges, so they do not form block-atoms.

We proceed with the reductions in H_r further till we obtain a primitive quotient graph H_s , for some $s \geq r$; see Figure 5.2. Notice that all atoms in H_r, \dots, H_{s-1} are necessarily block atoms since otherwise G_r would contain some proper atoms or dipoles and it would not be primitive. We add the newly discovered atoms to the catalog.

Now, the graph H_s consists of the core together with some pendant edges, loops and half-edges. Let H_0, \dots, H_{s-1} be the graphs obtained by an expansion series of H_s using Proposition 4.13. Notice that the core is preserved as an articulation/block in all these graphs. The core can be only changed by replacing of its colored edges by edge-quotients. Then the core in H_0 has to correspond to some block or articulation of H . We test all possible positions of the core in H . (We have $\mathcal{O}(n)$ possibilities, so we run the dynamic programming algorithm multiple times.) In what follows, we have the core fixed in H as well.

Overview of Dynamic Programming. Our goal is to apply a reduction series on H defining $\mathcal{H}_0, \dots, \mathcal{H}_t$. As already discussed above, we do not know which parts of G project to different parts of H . Therefore each \mathcal{H}_i is a set of graphs, and \mathcal{H}_t is a set of primitive graphs. We

5.1. Testing Expandability Using Dynamic Programming

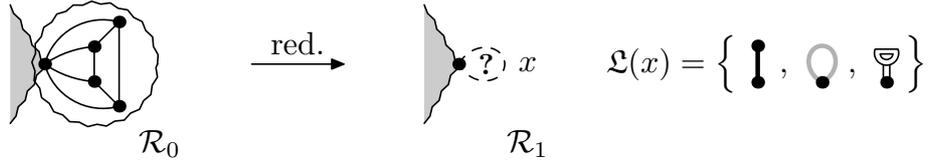


Figure 5.3: Let x be the pendant element corresponding to the pendant block of H depicted in Figure 5.1. Then $\mathfrak{L}(x)$ contains three different elements if all three atoms depicted in Figure 5.1 are contained in the catalog.

then determine expandability of H_r by testing whether $H_s \in \mathcal{H}_t$.

Since each set \mathcal{H}_i can contain a huge number of graphs, we represent it implicitly in the following manner. Each \mathcal{H}_i is represented by one graph \mathcal{R}_i with some colored edges and with so-called *pendant elements* attached to some vertices. Here, each pendant element can represent a pendant edge, loop or half-edge at the same time. Further for each pendant element x , we have a *list* $\mathfrak{L}(x)$ of possible realizations of the corresponding subgraph of H by the quotients from the catalog. Each graph of \mathcal{H}_i is created for \mathcal{R}_i by replacing the pendant elements by some edges, loops and half-edges from the respective lists.

Pendant elements of \mathcal{R}_i correspond to block parts of H with pairwise disjoint interiors. (Recall that block parts are defined in Section 3.) A pendant element x contains an edge/loop/half-edge in $\mathfrak{L}(x)$ if and only if it is possible to expand this edge/loop/half-edge to a graph isomorphic to the given block part. Further for each element of the list, we remember how to do this expansion. For an example, see Figure 5.3.

Testing whether $H_s \in \mathcal{H}_t$ is equivalent to testing whether there exists an *embedding* $H_s \hookrightarrow \mathcal{R}_t$. Here, the embedding is an isomorphism $\pi : H_s \rightarrow \mathcal{R}_t$ which maps pendant edges, loops and half-edges of H_s to the pendant elements of \mathcal{R}_t . Further, we require that the list of the pendant element $\pi(e)$ contains the mapped edge e .

These lists are used in the dynamic programming to compute \mathcal{R}_{i+1} from \mathcal{R}_i . According to Lemma 5.1, we have polynomially many atoms, and so the size of each list is polynomial. We note that one list may contain many half-edges.

Lemma 5.2. *Each list contains at most one edge and at most one loop. Further, if two lists contain the same edge or loop, then they have to be equal.*

Proof. If the pendant element of \mathcal{R}_i is fully expanded, it corresponds to one block part of H . Suppose that an edge- or a loop-quotient is in the list. If it is fully expanded, then it has to be isomorphic to this block part. But according to Lemma 4.10, there is only one way how to expand an edge- or a loop-quotient, because it can never contain half-edges. \square

Reductions with Lists. Suppose that we know \mathcal{R}_i , and we want to apply one step of the reduction and compute \mathcal{R}_{i+1} . First, we find all atoms in \mathcal{R}_i . (We define atoms with respect to the core block, and we consider pendant elements as pendant edges.) To construct \mathcal{R}_{i+1} from \mathcal{R}_i :

- We replace dipoles and proper atoms by edges of the corresponding colors from the catalog. If the corresponding dipole or proper atom is not contained in the catalog, we halt the reduction procedure.
- We replace block atoms by pendant elements with constructed lists. If some list is empty, we again halt the reduction.

It remains to describe the construction of the lists for the created pendant elements.

Let A be an atom in \mathcal{R}_i , replaced by an edge/pendant element e , and we want to compute the list for e . We call an atom A as a *star atom* if it consists of an articulation with attached pendant edges, loops and pendant elements.

Lemma 5.3. *Let A be a non-star block atom in \mathcal{R}_i . Then we can compute its list $\mathfrak{L}(A)$ in polynomial time.*

Proof. We iterate over all quotients in the catalog. For one such quotient Q , we determine whether $Q \hookrightarrow A$ where ∂Q is mapped to ∂A as follows. Notice that A is essentially 3-connected, and thus Q has to be 3-connected as well. (Otherwise an embedding does not exist.) So by (P3), we can test in polynomial time whether $Q \hookrightarrow A$ by coding colors of the pendant edges of Q by the colors of the vertices, and the lists of the pendant elements by lists of colors for vertices of A . If $Q \hookrightarrow A$, we add the edge/loop/half-edge representing this quotient to the list, and we remember the constructed mapping $Q \hookrightarrow A$. See Figure 5.4 for an example. \square

5.1. Testing Expandability Using Dynamic Programming

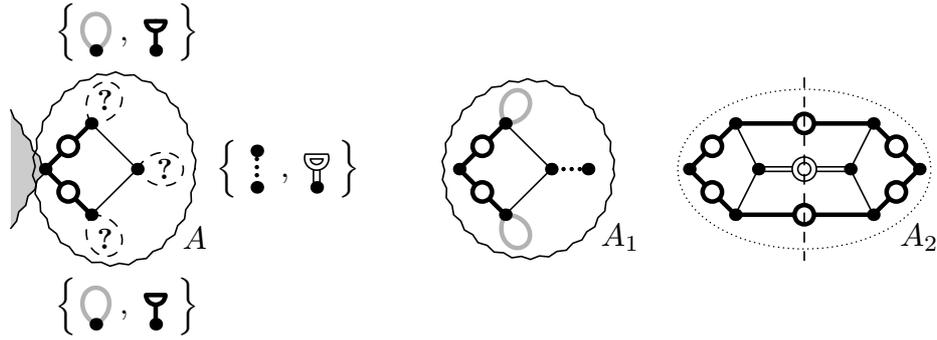


Figure 5.4: On the left, a non-star block atom A in \mathcal{R}_i with lists of its pendant elements depicted. On the right, two possible quotient which can be embedded into A . So the list of the pendant element replacing A in \mathcal{R}_{i+1} contains a pendant edge corresponding to the block atom A_1 and a half-edge corresponding to a half-quotient of the proper atom A_2 .

Lemma 5.4. *Let A be a star atom in \mathcal{R}_i . Then we can compute its list $\mathfrak{L}(A)$ in time $\mathcal{O}^*(2^{e(H)/2})$ where $e(H)$ is the number of edges in H .*

Proof. Each star atom of \mathcal{R}_i corresponds either to a block atom isomorphic to a star, or to the loop-quotient or a half-quotient of a dipole. Star atoms involve half-quotients of dipoles, and Lemma 4.12 states that a dipole can have exponentially many pairwise non-isomorphic half-quotients. For a dipole, we iterate over all possible half-quotients which gives $2^{e(H)/2}$ part in the complexity bound.

Case 1: Dipoles. First, we show how to deal with dipoles. We iterate over all dipoles in the catalog and try to add them to the list. For each dipole, we first test whether the lists of the pendant elements attached to the star atom A are compatible with the unique loop quotient. Then we iterate over all half-quotients of D . Let D be one dipole in the catalog with $\partial D = \{u, v\}$ and let Q be one of its at most $2^{e(H)/2}$ possible quotients. Recall from the proof of Lemma 4.12 that an edge of a dipole either projects to a half-edge, or two edges of the same color and type project to one loop. So each Q consists of loops and half-edges attached to u , and they have to be matched to the pendant elements of A with the corresponding lists.

We can reduce this problem to finding a perfect matching in bipartite graphs: Here, one part is formed by the loops and the half-edges of Q , and the other part is formed by the pendant elements of A . A loop/half-edge is adjacent to a pendant element, if and only if the

Chapter 5. Meta-algorithm

corresponding list contains this loop/half-edge. Each perfect matching defines one embedding $Q \hookrightarrow A$. We add the half-edge of the dipole D to the list if there exists a perfect matching.

Case 2: Star Atoms. We iterate over all star atoms of the catalog, let S be one of them. The star atom S consists of pendant edges, loops and half-edges attached to one vertex. Some of these half-edges correspond to dipoles, and some to proper atoms. Let e_1, \dots, e_d be the half-edges corresponding to the dipoles D_1, \dots, D_d . We construct all quotients Q of S by replacing e_1, \dots, e_d by all possible choices of the half-quotients Q_1, \dots, Q_d of D_1, \dots, D_d . In total, we have at most $2^{e(H)/2}$ different quotients Q of S . For each Q , we test by the matching procedure described above whether the edge representing the star atom S should be added to the constructed list. If yes, we add S to the list.

The procedure computes the list correctly since we test all possible quotients from catalog, and for each quotient we test all possibilities how it could be matched to A . For each quotient Q , the running time is clearly polynomial, and we have at most $2^{e(H)/2}$ quotients. \square

Algorithm 1 gives the pseudocode for computation of the list of a pendant element replacing an atom A . If the returned list is empty, we halt the reduction. Either H_r is not expandable to H , or we have chosen a wrong core in H . The following diagram shows the overview of the meta-algorithm:

$$\begin{array}{ccccccccccc}
 G_0 & \xrightarrow{\text{red.}} & G_1 & \xrightarrow{\text{red.}} & \cdots & \xrightarrow{\text{red.}} & G_r & & & & \\
 \Gamma_0 \downarrow & & \Gamma_1 \downarrow & & & & \Gamma_r \downarrow & & & & \\
 H_0 & \xleftarrow{\text{exp.}} & H_1 & \xleftarrow{\text{exp.}} & \cdots & \xleftarrow{\text{exp.}} & H_r & \xrightleftharpoons[\text{exp.}]{\text{red.}} & H_{r+1} & \xrightleftharpoons[\text{exp.}]{\text{red.}} & \cdots & \xrightleftharpoons[\text{exp.}]{\text{red.}} & H_s \\
 & & & & & & & & & & & & \downarrow ? \\
 \mathcal{R}_0 & \xrightarrow{\text{red.}} & \mathcal{R}_1 & \xrightarrow{\text{red.}} & \mathcal{R}_2 & \xrightarrow{\text{red.}} & \mathcal{R}_3 & \xrightarrow{\text{red.}} & \mathcal{R}_4 & \xrightarrow{\text{red.}} & \cdots & \xrightarrow{\text{red.}} & \mathcal{R}_t
 \end{array} \tag{5.1}$$

Lemma 5.5. *We can test whether $H_s \hookrightarrow \mathcal{R}_t$ in polynomial time.*

Proof. The graph H_s consists of the core together with edges, loops and half-edges attached to it. The graph \mathcal{R}_t consists of the core together with pendant elements with computed lists attached to it. Similarly

5.1. Testing Expandability Using Dynamic Programming

Algorithm 1 The subroutine for computing lists of realizations

Require: An atom A of \mathcal{R}_i .

Ensure: The list $\mathfrak{L}(x)$ of the pendant element x replacing A in \mathcal{R}_{i+1} .

- 1: Initiate the empty list $\mathfrak{L}(x)$.
 - 2: **if** A is a non-star atom **then**
 - 3: Iterate over all quotients from the catalog.
 - 4: For each quotient Q , apply (P3) to test whether $Q \hookrightarrow A$.
 - 5: If some embedding exists, we add the edge/loop/half-edge of Q to $\mathfrak{L}(x)$ together with the embedding.
 - 6: **end if**
 - 7: **if** A is a star atom **then**
 - 8: Iterate over all dipoles and star atoms in the catalog.
 - 9: **for** each dipole D **do**
 - 10: Test whether the loop-quotient of D matches the lists, if yes add the loop representing D to $\mathfrak{L}(x)$.
 - 11: Iterate over all half-quotients Q of D .
 - 12: **for** each half-quotient Q **do**
 - 13: Test existence of a perfect matching between loops and half-edges of Q and the pendant elements of A .
 - 14: If a perfect matching exists, add the half-edge of D to $\mathfrak{L}(x)$ together with its embedding, and proceed with the next dipole.
 - 15: **end for**
 - 16: **end for**
 - 17: **for** each star atom S **do**
 - 18: Compute all quotients Q by replacing the edges e_1, \dots, e_d corresponding to the dipoles by all possible combinations of half-quotients Q_1, \dots, Q_d .
 - 19: **for** each quotient Q **do**
 - 20: Test existence of a perfect matching exactly as above.
 - 21: If it exists, add the edge of S to $\mathfrak{L}(x)$ with its embedding, and proceed with the next star atom.
 - 22: **end for**
 - 23: **end for**
 - 24: **end if**
 - 25: **return** The constructed list $\mathfrak{L}(x)$.
-

as in the proof of Lemma 5.3, since the core of \mathcal{R}_t is essentially 3-connected, we use (P3) and test whether $H_s \hookrightarrow \mathcal{R}_t$. \square

The following lemma states that we can test the expandability of H_r using \mathcal{R}_t .

Lemma 5.6. *We have $H_s \hookrightarrow \mathcal{R}_t$ for some choice of the core in H if and only if H_r is expandable to H_0 which is isomorphic to H .*

Proof. First suppose that $H_s \hookrightarrow \mathcal{R}_t$ for some choice of the core. Then the embedding of H_s gives a sequence of replacements of edges, loops and half-edges by edge-quotients, loop-quotients and half-quotients respectively such that the resulting graph is isomorphic to H . We can apply these replacements in any order, since they modify the graph independently.

Therefore, we first replace pendant edges by the unique block atoms in H_s to get H_r which has to be compatible with the sequence of replacements defined by $H_s \hookrightarrow \mathcal{R}_t$. We do replacements in the manner of Proposition 4.13, and construct the expansions H_{r-1}, \dots, H_0 . Since we expand according to the embedding $H_s \hookrightarrow \mathcal{R}_t$, the constructed H_0 is isomorphic to H .

On the other hand, suppose that H_r is expandable to H_0 which is isomorphic to H . Then according to Lemma 4.15, the core of H_r is preserved in H , so it has to correspond to some block or to some articulation of H . We claim that $H_s \hookrightarrow \mathcal{R}_t$ for this choice of the core. There exists a sequence of replacements from H_s which constructs H_0 , and this sequence of replacements have to be possible in \mathcal{R}_t . Thus $H_s \hookrightarrow \mathcal{R}_t$. \square

5.2 Proof of The Main Theorem

Now, we are ready to establish the main algorithmic result of the paper; see Algorithm 2 for the pseudocode.

Theorem 1.1. We recall the main steps of the algorithm and discuss their time complexity. The reduction series G_0, \dots, G_r can be computed in polynomial time. The property (P2) ensures that there are polynomially many semiregular subgroups Γ_r of $\text{Aut}(G_r)$ which can be

Algorithm 2 The meta-algorithm for REGULARCOVER

Require: A graph G of \mathcal{C} satisfying (P0), (P1), (P2) and (P3), and a graph H .

Ensure: A regular covering projection $p : G \rightarrow H$ if it exists.

- 1: Compute the reduction series G_0, \dots, G_r ending with the unique primitive graph G_r .
 - 2: During the reductions, we construct the catalog by introducing all detected atoms and their quotients.
 - 3: Using (P2), we compute all semiregular subgroups Γ_r of $\text{Aut}(G_r)$.
 - 4: **for** each semiregular Γ_r **do**
 - 5: Compute the quotient $H_r = G_r/\Gamma_r$.
 - 6: Choose, say, the central block/articulation of H_r as the core.
 - 7: Compute the reduction series H_r, \dots, H_s with respect to the core.
 - 8: Introduce newly discovered atoms to the catalog.
 - 9: **for** each guessed position of the core in H **do**
 - 10: Compute the reduction series $\mathcal{R}_0, \dots, \mathcal{R}_t$ as follows.
 - 11: **for** each block atom A in \mathcal{R}_i **do**
 - 12: Compute its list using Algorithm 1.
 - 13: **end for**
 - 14: To construct \mathcal{R}_{i+1} , replace the proper atoms and dipoles of \mathcal{R}_i by colored edges, and the block atoms by pendant elements with the computed lists.
 - 15: **end for**
 - 16: Test whether $H_s \hookrightarrow \mathcal{R}_t$ by trying all possible mappings of the core of H_s to the core of \mathcal{R}_t .
 - 17: If yes, use the embedding to compute the expansions H_{s-1}, \dots, H_0 such that $H_0 \cong H$. And construct the groups $\Gamma_{r-1}, \dots, \Gamma_0$.
 - 18: The group $\Gamma := \Gamma_0$ defines the regular covering projection $p : G \rightarrow H$.
 - 19: **end for**
 - 20: **return** The regular covering projection p if it is constructed, “no” otherwise.
-

computed in polynomial time. (If G_r is an edge or a cycle, then it is clearly true as well.)

We compute the quotient $H_r = G_r/\Gamma_r$, and we fix the central block/articulation of H_r as the core. Then we compute the reduction series H_r, \dots, H_s by replacing only block atoms. We compute \mathcal{R}_t for all choices of the core in time $\mathcal{O}^*(2^{e(H)/2})$ and we can test $H_s \hookrightarrow \mathcal{R}_t$ in polynomial time according to Lemma 5.5. If we succeed for at least one choice of the core, then G regularly covers H . To certify this, we construct the regular covering projection as follows: We proceed as in the proof of Lemma 5.6 and construct $\Gamma_{r-1}, \dots, \Gamma_0$. Then $\Gamma := \Gamma_0$ gives a regular covering projection $p : G \rightarrow H$. If $H_s \not\hookrightarrow \mathcal{R}_t$ for all choices of the core, we proceed with the next choice of Γ_r . If we fail for all choices of Γ_r , the algorithm outputs “no”.

It remains to argue correctness of the algorithm. First suppose that the algorithm succeeds. According to Lemma 5.6 and Proposition 4.13, we construct a semiregular subgroup Γ of $\text{Aut}(G)$ such that $G/\Gamma \cong H$ which proves that G regularly covers H . On the other hand, suppose that there exists a semiregular Γ such that $H \cong G/\Gamma$. According to Lemma 4.7, it corresponds to the unique semiregular Γ_r on G_r which is one of the semiregular subgroups tested by the algorithm. Therefore H_r has to be expandable to H_0 isomorphic to H , and we detect this correctly according to Lemma 5.6. \square

Corollary 1.3. If G is 3-connected, then it is primitive and only block atoms can appear. So the expansion runs in polynomial time. If $|\Gamma| = |\Gamma_r|$ is odd, no half-edges occur in H_r , and so according Corollary 4.14 the expansion gives the unique graph H_0 . We can just test whether $H_0 \cong H$, or we can compute the reduction series $\mathcal{R}_0, \dots, \mathcal{R}_t$ while ignoring half-quotients. \square

5.3 More Details about Star Atoms and Their Lists

In this section, we give details and insides on lists of star atoms in \mathcal{R}_i . We show that this problem can be reduced to finding a certain generalization of a matching which we call IV-MATCHING. Here we describe a complete derivation of this problem, and in Conclusions we just give its combinatorial statement.

5.3. More Details about Star Atoms and Their Lists

An instance of the problem is depicted in Figure 5.5. Suppose that \mathcal{R}_i contains a star atom A with attached pendant elements, each with a previously computed list. We want to determine the list $\mathfrak{L}(A)$ which consists of some star atoms and half-quotients of dipoles from the catalog. Let S be a star atom from catalog, with attached half-edges, loops and pendant edges. We want to test whether S belongs to $\mathfrak{L}(A)$.

Before we do so, we establish basic properties of atoms concerning sizes. These properties allow us to understand the structure.

Size Properties. Let A be an atom and let Q be a quotient of this atom. We get the following relations between the sizes of Q and A , depending on the type of the quotient:

- Q is the edge-quotient: Then $v(Q) = v(A)$ and $e(Q) = e(A)$.
- Q is the loop-quotient: Then $v(Q) = v(A) - 1$ and $e(Q) = e(A)$.
- Q is a half-quotient: Then $v(Q) = v(A)/2$ and $e(Q) = e(A)/2$.

Throughout each reduction, we calculate how many vertices and edges are in all the atoms replaced by colored edges which we denote by \hat{v} and \hat{e} . Initially, we put $\hat{v}(e) = 0$ and $\hat{e}(e) = 1$ for every edge $e \in E(G_0)$. For a subgraph X , we then define

$$\hat{v}(X) := v(X) + \sum_{e \in E(X)} \hat{v}(e), \quad \text{and} \quad \hat{e}(X) := \sum_{e \in E(X)} \hat{e}(e).$$

When an atom A is replaced by an edge e in the reduction, we put $\hat{v}(e) = \hat{v}(\overset{\circ}{A})$ and $\hat{e}(e) = \hat{e}(\overset{\circ}{A})$. So for a subgraph X of G_i , the numbers $\hat{v}(X)$ and $\hat{e}(X)$ are the numbers of vertices and edges when X is fully expanded.

We similarly define \hat{v} and \hat{e} for quotients and their subgraphs; the difference here is that the quotients might contain half-edges. Let $H(X)$ be the set of half-edges of a subgraph X . Then for a half-edge $h \in H(X)$, created by halving an edge e , we put $\hat{v}(h) = \hat{v}(e)/2$ and $\hat{e}(h) = \hat{e}(e)/2$. For a subgraph X , we naturally define

$$\hat{v}(X) := v(X) + \sum_{e \in E(X)} \hat{v}(e) + \sum_{h \in H(X)} \hat{v}(h),$$

and

$$\hat{e}(X) := \sum_{e \in E(X)} \hat{e}(e) + \sum_{h \in H(X)} \hat{e}(h).$$

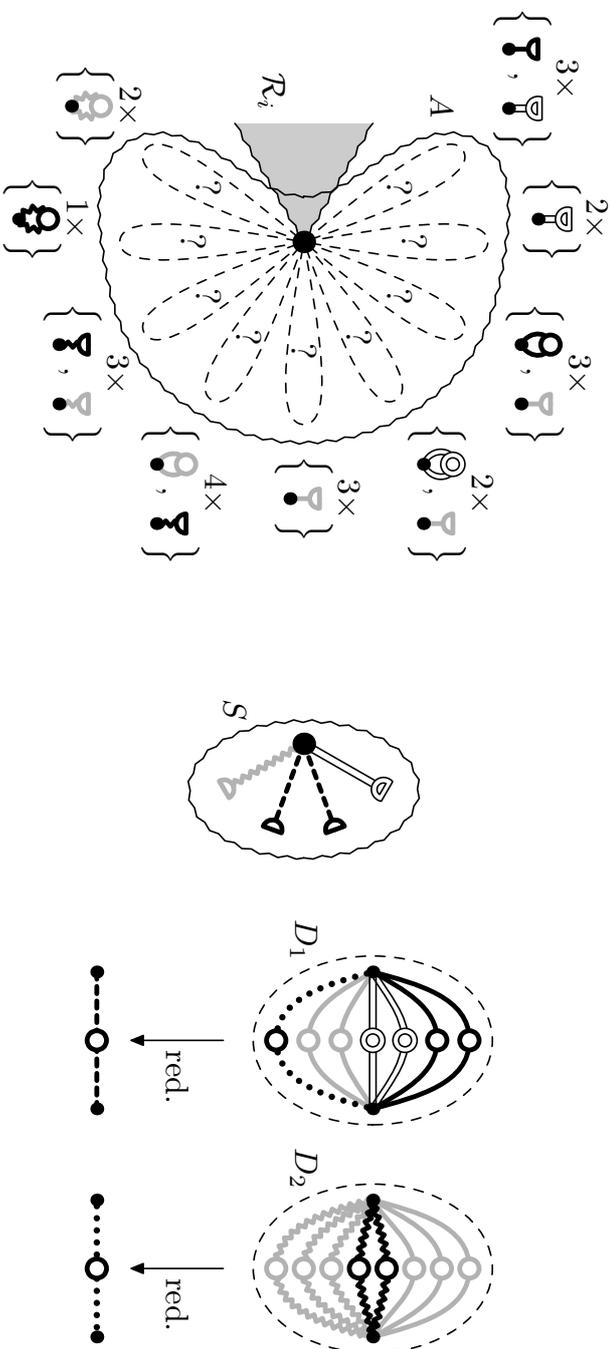


Figure 5.5: On the left, a star atom A in \mathcal{R}_i with 23 attached pendant elements, together with their lists and multiplicities. On the right, a star atom S from the catalog which belongs to $\mathcal{S}(A)$. The bold dashed edge corresponds to the dipole D_1 . The bold dotted edge in D_1 expands to another dipole D_2 . The remaining colored edges correspond to proper atoms.

5.3. More Details about Star Atoms and Their Lists

Finally, we also use \hat{v} and \hat{e} for pendant elements, created in the reduction series $\mathcal{R}_0, \dots, \mathcal{R}_t$.

We get straightforwardly the following:

Lemma 5.7. *For every pendant element x , the possible pendant edge, the possible loop and all half-edges of the list $\mathfrak{L}(x)$ have the same sizes \hat{v} and \hat{e} as $\hat{v}(x)$ and $\hat{e}(x)$ respectively. \square*

We apply this when $\mathfrak{L}(x)$ is computed since we consider only quotients of the correct sizes. Thus we can speedup the subroutine of Algorithm 1. But for purpose of this section, the following corollary is important:

Corollary 5.8. *Let x and y be two pendant elements.*

- (i) *If $\mathfrak{L}(x)$ and $\mathfrak{L}(y)$ contain a half-edge in common, then $\hat{v}(x) = \hat{v}(y)$ and $\hat{e}(x) = \hat{e}(y)$.*
- (ii) *Let $\mathfrak{L}(x)$ contains a loop of a color c and a half-edge of a color c' . Then $\mathfrak{L}(y)$ cannot contain both the loop of the color c' and the half-edge of the color c .*

Proof. (i) Implied by Lemma 5.7 directly.

(ii) Let $\mathfrak{L}(x)$ contain a loop e and $\mathfrak{L}(y)$ contain a half-edge h of the same color. Then $\hat{v}(x) = \hat{v}(e) + 1$ and $\hat{v}(y) = \hat{v}(e)/2 + 1$ for the vertices, and $\hat{e}(x) = \hat{e}(e)$ and $\hat{e}(y) = \hat{e}(e)/2$ for the edges. Therefore x is larger than y . For the same reason, we deduce that y is larger than x , which gives a contradiction. \square

The property (i) relates half-edges together. For pendant edges and loops, recall also Lemma 5.2. The property (ii) states that there is a certain size hierarchy on the pendant elements as we discuss below. We use this hierarchy to simplify the testing problem for A and S as follows.

Chains of Pendant Elements. Suppose that we ignore pendant edges contained in the lists since they are easy to deal with. Pendant elements are organized into independent chains, consisting of several levels. Each chain starts with pendant elements x of the level zero with $\hat{v}(x) = \alpha$ and $\hat{e}(x) = \beta$. Further, it contains on the level $m > 0$ the

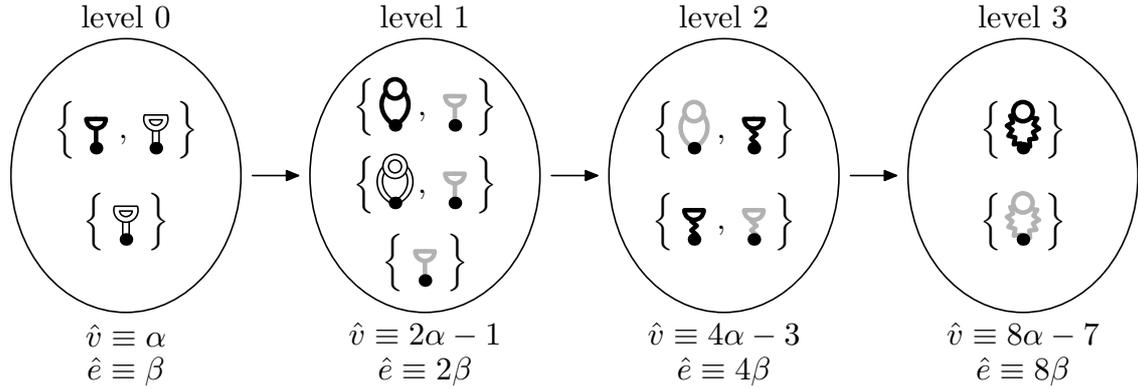


Figure 5.6: A chain of pendant elements with four levels, obtained from the example in Figure 5.5, for some α and β . Notice that quotients corresponding to one atom are placed in neighboring levels. All pendant elements of A are placed in this one chain, and we ignore their multiplicities.

pendant elements x with $\hat{v}(x) = 2^m\alpha - (2^m - 1)$ and $\hat{e}(x) = 2^m\beta$. See Figure 5.6 for an example of one chain from Figure 5.5.

A star atom A can contain multiple chains. It is important that different chains contain completely different colors in their lists, so they behave completely independently. If $\mathfrak{L}(x)$ contains a half-edge of a color c and $\mathfrak{L}(y)$ contains the loop of the same color c , then x belongs to a level m and y belongs to the level $m + 1$ of the same chain.

Preprocessing Dipoles. We have a star atom S with several pendant edges, loops and half-edges attached and we want to test whether it embeds into a star atom A in \mathcal{R}_i . Let us denote the single vertex of both S and A by v . As we discuss below, it is easy to deal with pendant edges and loops since they correspond to unique edge- and loop-quotients. On the other hand, the half-edges are more complex since we can have many different half-quotients of a half-edge. A half-edge can be of two types: Either it corresponds to a half-quotient of a proper atom, or of a dipole; for example in Figure 5.5 we have two half-edges corresponding to proper atoms, and two half-edges corresponding to dipoles. In the case of a proper atom, it corresponds to exactly one pendant element attached in A to v . (Alternatively, one subtree of blocks attached to v in H .) In the case of a dipole, a half-quotient of this dipole may correspond to several different pendants elements of A .

Recall Lemma 4.12 describing the structure of every half-quotient

5.3. More Details about Star Atoms and Their Lists

of a dipole. The resulting quotient has half-edges and loops attached to v . Again, edges of the dipole can be of two types:

- An edge can correspond to a proper atom, or alternatively it can be an original edge of G . Then in a half-quotient we obtain a half-edge/loop from this edge which corresponds after the full expansion to a subtree of blocks attached to v , and thus it corresponds to exactly one pendant element of A .
- Further, each dipole can also contain an edge corresponding to a dipole; for example in Figure 3.2, after one reduction step, we obtain on top a dipole consisting of two parallel edges, one corresponding to a dipole and the other to a proper atom. A half-quotient of this dipole would correspond to several pendant elements of A which we want to avoid. According to the definition of a dipole, there can be at most one edge corresponding to a dipole since every dipole contains all parallel edges between the given two vertices. Therefore, we can just expand this edge by replacing it with the edges of the dipole. We obtain exactly the same half-quotients as before.

Further, S may contain multiple half-edges corresponding to half-quotients of dipoles; this can be obtained by factorization as depicted in Figure 5.7. Suppose that each dipole is expanded as described, so it consists of some edges, each corresponding to a proper atom or an original edge of G . We would like to unify these dipoles into one dipole D , containing all the edges. But this might introduce additional

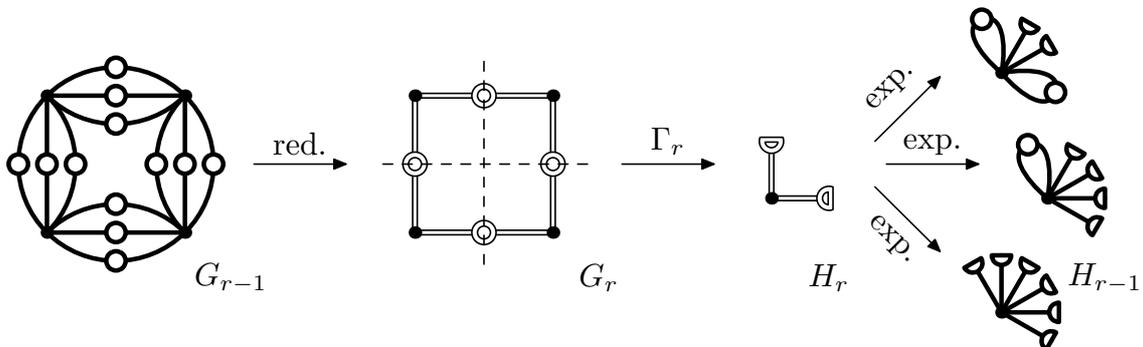


Figure 5.7: By two reflections, the quotient H_r consists of a star atom with two half-edges corresponding to dipoles. All three expansions H_{r-1} up to isomorphism are depicted. It is not possible to expand a quotient H_{r-1} with three attached loops.

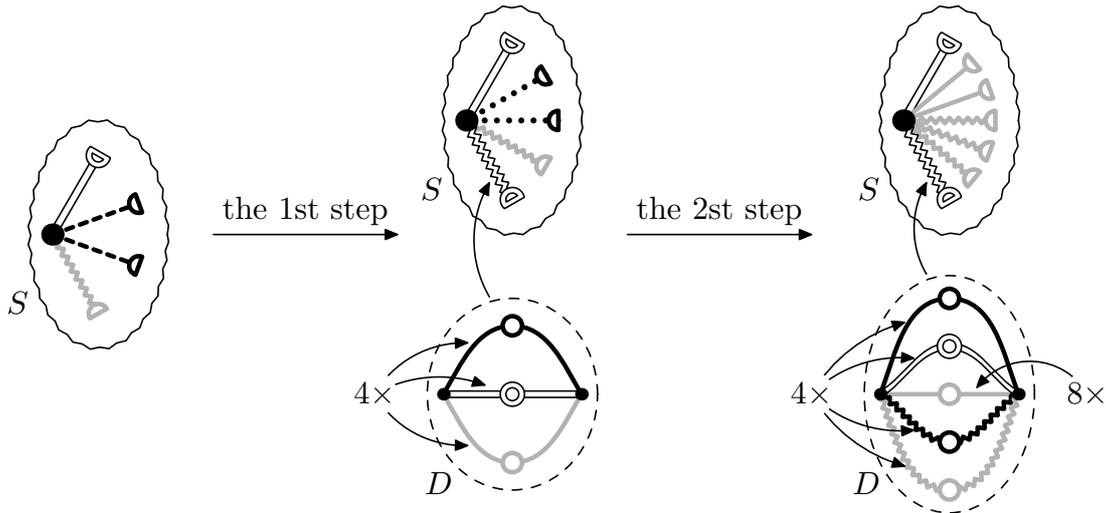


Figure 5.8: In the first step, we expand both dipoles D_1 , each having two edges of each of the three color classes corresponding to proper atoms. So the unified dipole D now contains four colored edges for each of the three color classes. Further D_1 has one halvable edge corresponding to a dipole D_2 , so one half-edge of this color is attached to S directly for each D_1 . In the second step, we expand the two half-edges corresponding to D_2 . Here we have two black curly edges, which are directly placed to D . But the remaining two color classes have both odd sizes, so one edge from each is directly attached as a half-edge to S . The remaining edges are placed to D . The resulting star atom S together with the resulting unified dipole D is depicted on the right.

quotients as in Figure 5.7. If two dipoles both contain an odd number of edges of one color, in the unified dipole we have a half-quotient consisting of only loops of this color which is not possible in the case of two separated dipoles. There is an easy fix of the problem, we preprocess each dipole and we remove one edge from each color class of odd size (of necessarily halvable edges) and place it as the half-edge directly in S attached to v . We surely know that at least one half-edge of this color appears in every half-quotient of this dipole. The resulting star atom S belongs to $\mathfrak{L}(A)$ if and only if the original S belongs there. In Figure 5.8, we illustrate this preprocessing for the example in Figure 5.5.

Dealing with Attached Edges and Loops. After the preprocessing step, the star atom S contains several pendant edges, loops, half-edge attached to v , with at most one half-edge corresponding to a dipole. If

5.3. More Details about Star Atoms and Their Lists

this dipole contains some non-halvable edges, then they are paired in every half-quotient and form loops. So we can remove them from the dipole and attach the corresponding number of loops directly to v in S . After this step, the dipole contains only even number of halvable edges in each color class.

Each pendant edge corresponds to a block atom attached to v . Each loop either corresponds to a proper atom, or a dipole. Similarly as before, we expand these dipoles, replacing them by several loops corresponding to their edges. After this expansion, each pendant edge/loop corresponds to exactly one pendant element of A . But since their expansion is unique, they can be contained in lists of only one type of pendant elements. Therefore we can arbitrarily assign pendant elements and remove these pendant elements and loops from S and the assigned pendant elements from A .

The resulting star atom S contains only half-edges and at most one half-edge corresponds to a dipole having only halvable edges corresponding to proper atoms. Hence the considered star atom decomposes into the half-edges H corresponding to proper atoms and at most one half-edge corresponding to a halvable dipole D .

Reduction to the V-Matching Problem. Suppose that S contains a half-edge corresponding to a dipole D , otherwise the half-quotient of S is unique and we can easily match it A using perfect matching in a bipartite graph, as described in the proof of Lemma 5.4. Let H be the set of the remaining half-edges attached in S corresponding to proper atoms. In each half-quotient of the dipole D , we have color classes of even sizes. For each color class of an even size m , we can choose an arbitrary number ℓ of loops and the corresponding number of half-edges $m - 2\ell$. If we know these values m and ℓ , we can just test existence of a perfect matching as described in Lemma 5.4. Since we do not know the values m and ℓ , we need to solve a generalization of perfect matching called IV-MATCHING in which we are free to choose these values.

The input of IV-MATCHING gives a bipartite graph B defined similarly as before. One part has a vertex per a pendant element and the other part has a vertex for each edge of D and half-edge of H . We put an edge between $e \in D$ and x , if a half-quotient of e or the loop-quotient created by rotating two edges e is in the list $\mathcal{L}(x)$. We call the first case a *half-incidence* and the latter case a *loop-incidence*.

Further, we add a *half-incidence* between $h \in H$ and x , if the half-edge h is contained in the list $\mathcal{L}(x)$.

We ask whether there exists a *spanning subgraphs* B' of B , with each component of connectivity a path of length one or two, as follows. Each pendant element x is in B' either half-incident to exactly one vertex in the other part, or it is loop-incident to exactly two edges $e, e' \in D$ of the same color class. Further, each edge and half-edge is incident in B to exactly one pendant element. In what follows, we call B' an *IV-subgraph* of B . See Figure 5.9 for an example, with several additional properties which we discuss now.

Special Properties of Inputs. The *IV-MATCHING* would likely be *NP-complete* in general, but there are additional properties already depicted in Figure 5.9. We describe them in details, since they might help in constructing a polynomial-time algorithm for this problem.

First, B consists of separate connected components and for each we can solve the problem separately. A connected component is induced by a chain of pendant elements and its incident vertices representing S . In what follows, we assume that we just have a single chain. This chain has its level structure which translates into a level structure for part with half-edges of H and edges of D as follows.

- Let h be a half-edge of H with $\hat{v}(h) = \alpha$ and $\hat{e}(h) = \beta$. Then this half-edge can be in B half-incident to pendant elements x only of the level with $\hat{v}(x) = \alpha$ and $\hat{e}(x) = \beta$.
- Let e be an edge of the dipole D with $\hat{v}(e) = 2\alpha$ and $\hat{e}(e) = 2\beta$. It can be half-incident to pendant elements x only of the level with $\hat{v}(x) = \alpha$ and $\hat{e}(x) = \beta$ and loop-incident to pendant elements x only of the level with $\hat{v}(x) = 2\alpha - 1$ and $\hat{e}(x) = \beta$.

So we also have levels for the other part of B . Let us call these levels *A levels* and *S levels*, respectively. If we depict all levels from left to right according to their order, alternating *A levels* of pendant elements levels and *S levels* of half-edges of H and edges of D . The graph B has only edges going between consecutive levels as depicted in Figure 5.9.

The second property says that B can be viewed as a cluster graph. In each *S level*, the edges of D and the half-edges of H form clusters according to their color classes. Similarly in each *A level*, the pendant elements form clusters according to equivalence classes of their

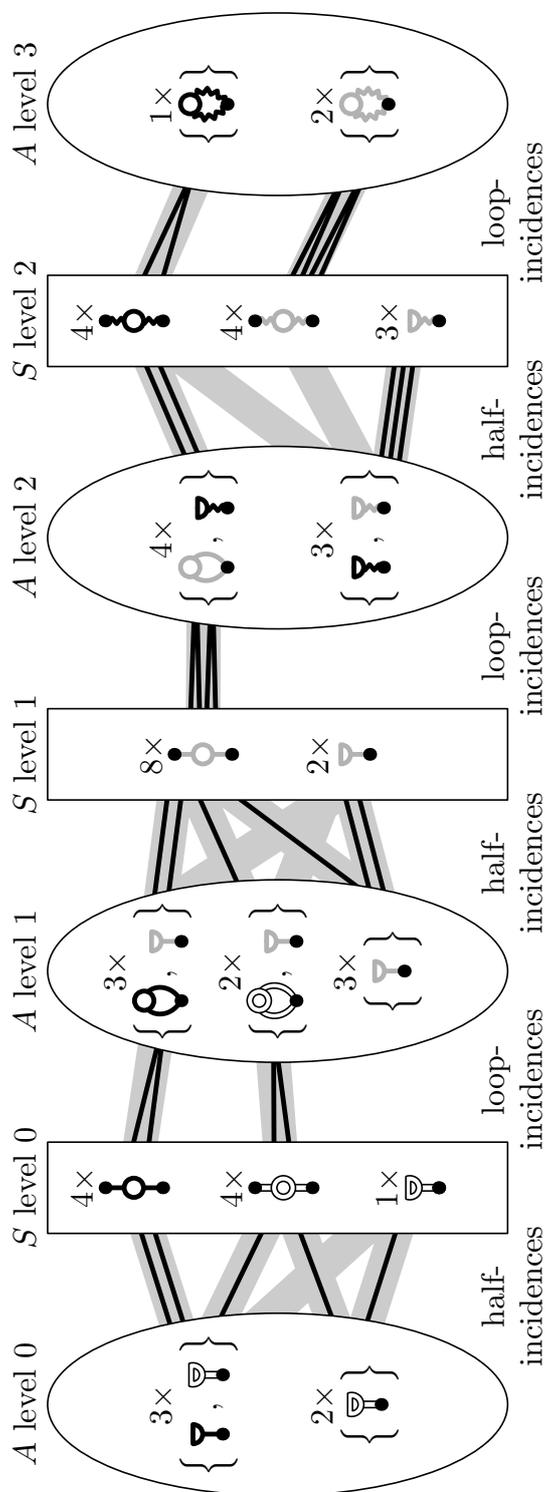


Figure 5.9: A bipartite graph B with edges depicted in gray and a spanning subgraph B' highlighted in bold. The part of pendant elements is in circles, the other part is in boxes. The edges of D are depicted by edges and the remaining half-edges corresponding to proper atoms and depicted by half-edges.

lists. (We note that two pendant elements with equal lists can correspond to non-isomorphic subgraphs in H . Then their lists contain only half-edges.) Two clusters from different levels are either completely adjacent, or not adjacent at all. In other words, the subgraph induced by all edges between two clusters is either complete bipartite, or contains no edges.

Also each edge cluster is loop-adjacent to at most one pendant element cluster, since a loop-quotient contained in a list uniquely determines the pendant element, as discussed above. We do not need the condition that each list is loop-incident in the IV-subgraph B' to two edges of the same color class since this is automatically achieved. On the other hand, there are no constraints for half-incidences between clusters.

The third property reads as follows. If the highest level is an A level, we can deal with this level greedily. All pendant elements of this level has to be loop-incident in B' , so we can ignore half-edges in their lists and assign them directly to the previous S level. For instance in Figure 5.9, we know that we have two black shaken loops and one gray shaken loops. Therefore we can remove this level together with the corresponding number of edges of D . We can assume that the highest level is an S level. Similarly if the smallest level is an S level, we can match it greedily. If the smallest level is an A level, this greedy approach does not work. We know how many lists have to be realized by their half-quotients. But the lists may contain multiple half-edge colors and therefore may be realized by different edges of D . Therefore in the next S level we may end up with different numbers of edges of D which may change solvability on higher levels.

Another property is that we know for each level of incidences, between two consecutive levels, how many edges have to be placed in B' . We can just process the graph B from left to right. On the smallest A level, we know how many half-incidences are in B' . Therefore we can compute the remaining number of edges of D which are loop-incident in B' . By processing in this way from left to right, we get all the numbers. Notice that we do not know how many half- and loop-incidences are in B' for each cluster, otherwise we could solve the problem directly by a perfect matching in a modified graph.

Since the sizes of the levels are growing exponentially from left to right, we have at most logarithmically many levels with respect to the

5.3. More Details about Star Atoms and Their Lists

size of the graph H . One can also show that the number of vertices in each level is somewhat limited by the size of the level, but it does not seem to give any useful bound.

6

Applying the Meta-algorithm to Planar Graphs

In this section, we show that the meta-algorithm described in Section 5 applies to the class of planar graphs. We describe the properties of planar graphs and their quotients in more details than necessary. We do it to give a deeper insight into regular covers of planar graphs.

6.1 Automorphism Groups of 3-connected Planar Graphs

We review the well-known properties of planar graphs and their automorphism groups. These strong properties are based on the Whitney's Theorem [41] stating that a 3-connected graph has a unique embedding into the sphere.

Spherical Groups. A group is *spherical* if it is a group of symmetries of a tiling of the sphere. The first class of spherical groups are the subgroups of the automorphism groups of the platonic solids, i.e., \mathbb{S}_4 for the tetrahedron, $\mathbb{C}_2 \times \mathbb{S}_4$ for the cube and the octahedron, and $\mathbb{C}_2 \times \mathbb{A}_5$ for the dodecahedron and the icosahedron. See Table 6.1 for the number of conjugacy classes of the subgroups of these three groups. Note that conjugate subgroups Γ determine isomorphic quotients G/Γ . The second class of spherical groups is formed by infinite families \mathbb{C}_n and \mathbb{D}_n .

Automorphisms of a Map. A map \mathcal{M} is a 2-cell embedding of a graph G onto a surface S . For purpose of this paper, S is either the sphere or the projective plane. A *rotational scheme* at a vertex is a

| S_4 of the order 24 | | | |
|-----------------------|--------|-------|--------|
| Order | Number | Order | Number |
| 1 | 1 | 6 | 1 |
| 2 | 2 | 8 | 1 |
| 3 | 1 | 12 | 1 |
| 4 | 3 | | |

| $C_2 \times S_4$ of the order 48 | | | |
|----------------------------------|--------|-------|--------|
| Order | Number | Order | Number |
| 1 | 1 | 8 | 7 |
| 2 | 5 | 12 | 2 |
| 3 | 1 | 16 | 1 |
| 4 | 9 | 24 | 3 |
| 6 | 3 | | |

| $C_2 \times A_5$ of the order 120 | | | |
|-----------------------------------|--------|-------|--------|
| Order | Number | Order | Number |
| 1 | 1 | 8 | 1 |
| 2 | 3 | 10 | 3 |
| 3 | 1 | 12 | 2 |
| 4 | 3 | 20 | 1 |
| 5 | 1 | 24 | 1 |
| 6 | 3 | 60 | 1 |

Table 6.1: The number of conjugacy classes of the subgroups of the groups of platonic solids.

cyclic ordering of the edges incident with the vertex. When working with abstract maps, they are graphs with a rotational scheme given for every vertex. An *angle* is a triple (v, e, e') where v is a vertex, and e and e' are two incident edges which are consecutive in the rotational scheme of v .

An automorphism of a map is an automorphism of the graph which preserves the angles; in other words the rotational scheme is preserved up to reflections. With the exception of paths and cycles, $\text{Aut}(\mathcal{M})$ is a subgroup of $\text{Aut}(G)$. In general these two groups might be very different. For instance, the star S_n has $\text{Aut}(S_n) = S_n$, but for any map \mathcal{M} of S_n we just have $\text{Aut}(\mathcal{M}) = \mathbb{D}_n$.

Lemma 6.1. *For any map \mathcal{M} , a permutation representation of the*

6.2. Primitive Graphs and Atoms for Planar Graphs

group $\text{Aut}(\mathcal{M})$ can be computed in time $\mathcal{O}(m^2)$ where m is the number of edges of \mathcal{M} .

Sketch. There are $\mathcal{O}(m)$ angles in \mathcal{M} . We fix one angle (v, e, e') , and test for each other angle whether there is an automorphism mapping (v, e, e') to it. The key observation is that if such an automorphism exists, it is uniquely determined. We can just test in $\mathcal{O}(n+m)$ whether the map is compatible with this prescribed mapping. The total running time is $\mathcal{O}(m^2)$. \square

If \mathcal{M} is drawn on the sphere, then $\text{Aut}(\mathcal{M})$ is isomorphic to one of the spherical groups [23, 14]. In case G is a 3-connected planar graph, there exists the unique embedding of G onto the sphere. Then for any map \mathcal{M} of G , we have $\text{Aut}(G) \cong \text{Aut}(\mathcal{M})$ [41]. Thus we get the following corollary of Lemma 6.1.

Corollary 6.2. *If G is a 3-connected planar graph, then $\text{Aut}(G)$ is isomorphic to one of the spherical groups. We can determine this group in polynomial time and find permutations which generate it in time $\mathcal{O}(n^2)$.*

We note that a linear-time algorithm for computing automorphism groups of planar graphs is known [28].

6.2 Primitive Graphs and Atoms for Planar Graphs

To be able to apply the meta-theorem on planar graphs, we first investigate the automorphism groups of atoms and primitive graphs, and their quotients. Recall that a graph is essentially 3-connected if it is a 3-connected graph with attached single pendant edges.

Lemma 6.3. *For a planar primitive graph G , the group $\text{Aut}(G)$ is a spherical group and can be computed in time $\mathcal{O}(n^2)$.*

Proof. If G is essentially 3-connected, this directly holds according to Lemma 3.7 and Corollary 6.2. If G is K_2 or C_n with attached single pendant edges, then $\text{Aut}(G)$ can be computed trivially as well. \square

For an atom A , Recall that $\text{Fix}(A)$ is the point-wise stabilizer of ∂A in $\text{Aut}(A)$. Further, by $\text{Aut}_{\partial A}(A)$ we denote the set-wise stabilizer of ∂A in $\text{Aut}(A)$.

Lemma 6.4. *We get the following automorphism groups for atoms:*

- *For a star block atom, we get in general $\text{Aut}_{\partial A}(A) = \text{Fix}(A)$ which is a direct product of symmetric groups.*
- *For a dipole, we get in general $\text{Fix}(A)$ as a direct product of symmetric groups. For a symmetric dipole, we have $\text{Aut}_{\partial A}(A) = \text{Fix}(A) \rtimes \mathbb{C}_2$, for non-symmetric $\text{Aut}_{\partial A}(A) = \text{Fix}(A)$.*
- *For a proper atom A of a planar graph, the group $\text{Aut}_{\partial A}(A)$ is a subgroup of \mathbb{C}_2^2 and $\text{Fix}(A)$ is a subgroup of \mathbb{C}_2 .*
- *For a non-star block atom A of a planar graph, $\text{Aut}_{\partial A}(A) = \text{Fix}(A)$ and it is a subgroup of \mathbb{D}_n .*

Proof. The edges of the same color class of a star block atom A can be arbitrary permuted, so we get for $\text{Aut}_{\partial A}$ and $\text{Fix}(A)$ the same group which is a direct product of symmetric groups. For a dipole the situation is similar, just for a symmetric dipole we can permute the vertices in ∂A , so we get a semidirect product with \mathbb{C}_2 .

Let A be a proper atom with $\partial A = \{u, v\}$, and let A^+ be the essentially 3-connected graph created by adding the edge uv . Since $\text{Aut}_{\partial A}(A)$ preserves ∂A , we have $\text{Aut}_{\partial A}(A) = \text{Aut}_{\partial A}(A^+)$, and further $\text{Aut}_{\partial A}(A^+)$ fixes in addition the edge uv . Because A^+ is essentially 3-connected, then $\text{Aut}_{\partial A}(A^+)$ corresponds to the stabilizer of uv in $\text{Aut}(\mathcal{M})$ for the unique map \mathcal{M} of A^+ . But such a stabilizer has to be a subgroup of \mathbb{C}_2^2 . For $\text{Fix}(A)$ we further stabilize the vertices of ∂A , so it is a subgroup of \mathbb{C}_2 .

Let A be a non-star block atom. Then $\partial A = \{u\}$ is preserved, so we have one fixed vertex in both $\text{Aut}_{\partial A}(A)$ and $\text{Fix}(A)$, so they are the same. Since A is essentially 3-connected, then $\text{Aut}_{\partial A}(A)$ is a subgroup of \mathbb{D}_n where n is the degree of u . □ □

As a corollary of Proposition 4.4, we can characterize automorphism groups of planar graphs, similarly to Babai [4, 5].

Corollary 6.5. *Each automorphism group of a planar graph is obtained from a spherical groups by repeated semidirect products of direct products of groups of atoms from Lemma 6.4.*

Proof. The primitive graph G_r has a spherical automorphism group by Lemma 6.3. Recall $\text{Aut}(G_i) \cong \text{Aut}(G_{i+1}) \rtimes \text{Ker}(\Phi_i)$. The kernel

6.2. Primitive Graphs and Atoms for Planar Graphs

$\text{Ker}(\Phi_i)$ is a direct product of the groups $\text{Fix}(A)$ for all atoms A in G_i , which are by Lemma 6.4 subgroups of \mathbb{C}_2 and \mathbb{D}_n , and direct products of symmetric groups. \square

Geometry of Quotients. We review in more details the geometry of quotients of planar graphs. These are classical results from geometry of the sphere, and the reader can find the missing proofs and details in [40]. We note that this precise understanding is not necessary for the correctness of the meta-algorithm, but we believe it gives a deeper insight into the quotients of planar graphs, in the direction of the Negami's Theorem [37].

We start by basic definitions from geometry. An automorphism of a 3-connected planar graph is called *orientation preserving*, if the respective map automorphism preserves the global orientation of the surface. It is called *orientation reversing* if it changes the global orientation of the surface. A group of automorphisms of a 3-connected planar graph is either orientation preserving, or it contains a subgroup of index two of orientation preserving automorphisms. (The reason is that composition of two orientation reversing automorphisms is an orientation preserving automorphism.)

Let τ be an orientation reversing involution. The involution τ is called *antipodal* if it is a semiregular automorphism of a closed orientable surface S such that $S/\langle\tau\rangle$ is a non-orientable surface. Otherwise τ is called a *reflection*. A reflection of the sphere fixes a circle.

In particular, the half-quotient of the sphere by an antipodal involution is the projective plane and the half-quotient by a reflection is a disk. An orientation reversing involution of a 3-connected planar graph is called *antipodal* if the respective map automorphism is antipodal and it is called a *reflection* if the respective map automorphism is a *reflection*. A reflection of a map on the sphere fixes always either an edge, or a vertex.

Lemma 6.6 ([40]). *Let G be a 3-connected planar graph and Γ be a semiregular subgroup of $\text{Aut}(G)$. Then the following can happen:*

- (a) *The action of Γ is orientation preserving and the quotient G/Γ is planar,*

- (b) *The action of Γ is orientation reversing but does not contain an antipodal involution. Then the quotient G/Γ is planar and necessarily contains half-edges,*
- (c) *The action of Γ is orientation reversing and contains an antipodal involution. Then G/Γ is projective planar. \square*

As we show, if G/Γ is projective non-planar, then necessarily also the quotient of associated primitive graph G_r/Γ_r is projective non-planar. The reason is that a projection of an atom is always planar, and planarity is preserved by expansions.

Lemma 6.7. *Let A be a proper atom in planar graph and let $\partial A = \{u, v\}$. Then there are at most two involutory semiregular automorphisms $\tau \in \text{Aut}_{\partial A}(A)$ transposing u and v . Moreover, if there are exactly two of them, then one is orientation preserving and the other one is a reflection. In particular, the two respective quotient graphs may be non-isomorphic planar graphs.*

Proof. The graph A^+ is an essentially 3-connected planar graph with a unique embedding into the plane. Then any graph automorphism τ transposing u and v is a map automorphism fixing e . It is easy to see that either τ is a 180 degree rotation around the centre of e , or it is a reflection [4, 5]. According to Lemma 6.6, both possible quotients are planar. \square

The following lemma is straightforward, and completes the description of possible quotients of primitive graphs.

Lemma 6.8. *Let H be a quotient of a cycle C by a semiregular group of automorphisms. Then either H is a cycle, or a path, or a path with one pendant half edge, or H is a path with two pendant half edges. Depending on parity, just three of the above for cases happen. \square*

6.3 Planar Graphs Satisfy (P0) to (P3)

In this section, we prove that planar graphs satisfy the properties (P0) to (P3), and thus the meta-algorithm of Section 5 applies to them. The class of planar graphs clearly satisfies (P0). The graph isomorphism can be tested in polynomial time for planar graphs [28] and for projective planar graphs [32], which implies (P1). For (P2), Corollary 6.2

6.3. Planar Graphs Satisfy (P0) to (P3)

states that the automorphism group of a 3-connected planar graph is a spherical group, and so we can generate all at most linearly many subgroups of given order and check which ones act semiregularly. It remains to prove (P3):

Proposition 6.9. *The class of planar graphs satisfies the property (P3).*

Proof. Let H be a 3-connected projective planar graph. By Lichtenstein [32], there are two possible cases: Either the number of possible projective maps of H is bounded by $c \cdot v(H)$, or H contains an edge uv such that $H \setminus \{u, v\}$ is a planar graph. The property (P3) gives two 3-connected graphs G and H , and if $G \hookrightarrow H$, then necessarily $G \cong H$ (while ignoring the colors), and thus the same case happens for both G and H . We deal with these two cases separately.

Case 1: Linear number of projective maps. We want to show that the size of $\text{Aut}(H)$ is polynomial in $v(H)$. To see this, every $\pi \in \text{Aut}(H)$ is either a map automorphism, or an isomorphism between two maps of G . The number of automorphisms of a map is bounded by $4e(H)$. Since we have a linear number of non-isomorphic maps, we get the bound $4c \cdot e(H) \cdot v(H)$, which is $\mathcal{O}(v^2(H))$. We can construct all these automorphisms in polynomial time, for instance by attaching specific gadgets to the vertices of G and H , and testing whether the modified graphs are still isomorphic. Therefore we can compute $\text{Aut}(H)$ in polynomial time, and by composition with some isomorphism from G to H , we can test whether there exists an embedding $G \hookrightarrow H$.

Case 2: Removal of two vertices makes each of the graphs G and H planar. We know that there exists $uv \in E(G)$ and $u'v' \in E(H)$ such that $G \setminus \{u, v\}$ and $H \setminus \{u', v'\}$ are both planar graphs. We test $4e^2(G)$ possible choices of the pairs of vertices, and we want to test whether there exists an embedding π such that $\pi(u) = u'$ and $\pi(v) = v'$. Clearly, $c(u) \in \mathfrak{L}(u')$ and $c(v) \in \mathfrak{L}(v')$. It remains to deal with the planar remainders $G' = G \setminus \{u, v\}$ and $H' = H \setminus \{u', v'\}$. Notice that G' and H' are not necessarily 3-connected.

Now, we can approach the problem exactly as with testing expandability of H_r in the meta-algorithm. We code the colors of the vertices of G' by the colors of the single pendant edges attached to the corresponding vertices of G' , and we code the lists of the colors by the

Chapter 6. Applying the Meta-algorithm to Planar Graphs

list of the pendant elements attached to the vertices in H' . We choose the central block/articulation of both G' and H' as the core.

First, we proceed with a series of reductions G'_0, \dots, G'_r of G' , replacing the atoms by edges and placing them in the separate catalog. Here, the automorphisms are indeed color preserving. Further, we have four partition classes of $V(G')$ according to the adjacencies to u and v , and automorphisms preserves them. We end up with a primitive 3-connected graph G'_r .

Using the constructed catalog, we proceed similarly with the reductions $\mathcal{R}'_0, \dots, \mathcal{R}'_r$ of H' . But there are two important differences. First, the starting graph \mathcal{R}'_0 already contains pendant elements with initial lists, given by H' . The second difference is that we have to remember lists also for interior edges of \mathcal{R}'_i , not only for pendant elements. The reduction is done similarly, by replacing the atoms of \mathcal{R}'_i by edges and pendant elements in \mathcal{R}'_{i+1} . Let A be an atom in \mathcal{R}'_i , and to compute the list we iterate over all atoms of G' of the given type in the catalog. Let B be one such atom from the catalog. Depending on the type of A , we proceed as follows:

- If A is a block atom of \mathcal{R}'_i , then according to Lemma 6.4 we have $\text{Aut}_{\partial A}(A)$ is a subgroup of \mathbb{D}_n , and so we can test all possible isomorphisms from B to A . If one such isomorphism defines an embedding of B to A , we add the edge representing B to the list.
- If A is a proper atom of \mathcal{R}'_i , then we similarly test according to Lemma 6.4 at most four possible isomorphisms from B to A .
- If A is a dipole, we construct a bipartite graph. One part is formed by the colored edges of B , the other part by the edges of A with the associated lists of possible colors. The adjacencies are given by containments of the colors in the lists. We test existence of a perfect matching in this bipartite graph.

Finally, when we reach the primitive graph \mathcal{R}'_r , then according to Lemma 6.3 we have that $\text{Aut}(\mathcal{R}'_r)$ is a spherical group. Thus we can test all possible isomorphisms from G'_r to \mathcal{R}'_r whether they are compatible with the lists of \mathcal{R}'_r . The entire subroutine is clearly correct and runs in polynomial time. \square

Corollary 1.2. We just apply Theorem 1.1 since planar graphs satisfy (P0) to (P3). \square

7

Concluding Remarks

We conclude this paper by remarks to the meta-algorithm and open problems.

7.1 Possible Extensions of The Meta-algorithm

There are several possible natural extensions of the meta-algorithm. First, we can easily generalize it for input graph G and H with half-edges, directed edges and halvable edges, and also for colored graphs. Further, one can prescribe lists of possible images for the vertices $V(G)$ and of possible pre-images of the vertices $V(H)$, the expandability testing can compute also with these lists.

The most interesting of these extensions is to use the meta-algorithm to generate all quotients H of G . Indeed, this cannot be achieved in polynomial time since there might be exponentially many such quotients. But we can enumerate all labeled quotients with a polynomial delay, by using different expansions determined in Proposition 4.13.

7.2 Open problems

We already discussed that `REGULARCOVER` generalizes the graph isomorphism problem. We ask the following:

Problem 7.1. *Is the problem `REGULARCOVER` GI-complete?*

Chapter 7. Concluding Remarks

Similarly to the graph isomorphism problem, we are given two graphs G and H which restrict each other. To solve `REGULARCOVER`, one needs to understand automorphism groups of graphs and their semiregular subgroups.

As possible next direction of research, we suggest to attack classes of graphs close to planar graphs, for instance projective planar graphs or toroidal graphs. To do so, it seems that new techniques need to be built. Also the automorphism groups of projective planar graphs and toroidal graphs are not well understood.

Lastly, we indeed ask whether the slow subroutine for dipole expansion of the meta-algorithm can be solved in polynomial time:

Problem 7.2. *Can the complexity of the algorithm of Theorem 1.1 be improved to be polynomial?*

In Section 5.3, we have shown that the slow subroutine reduces to a generalized matching problem called `IV-MATCHING`. Here, we describe a purely combinatorial formulation of the `IV-MATCHING` problem. This reformulation can be useful to people studying combinatorial optimization since they can attack the problem without understanding the regular coverings and the structural results obtained in the entire paper.

The input of `IV-MATCHING` consists of a bipartite graph $B = (V, E)$ with a partitioning V_1, \dots, V_ℓ , with ℓ even, of its vertices which we call *levels*, with all edges between of consecutive levels V_i and V_{i+1} , for $i = 1, \dots, \ell - 1$. The levels V_1, V_3, \dots are called *odd* and the levels V_2, V_4, \dots *even*. Further each level V_i is partitioned into several *clusters*, each consisting of a few vertices with identical neighborhoods. There are three key properties:

- The incidences in B respect the clusters; between any two clusters the graph B induces either a complete bipartite graph, or an edge-less graph.
- Each cluster of an even level V_{2t} is incident with at most one cluster at the odd level V_{2t+1} , and each cluster of V_{2t+1} is incident with at most one cluster at V_{2t} ; so we have a matching between clusters at levels V_{2t} and V_{2t+1} .
- The incidences between the clusters of V_{2t-1} and V_{2t} can be arbitrary.

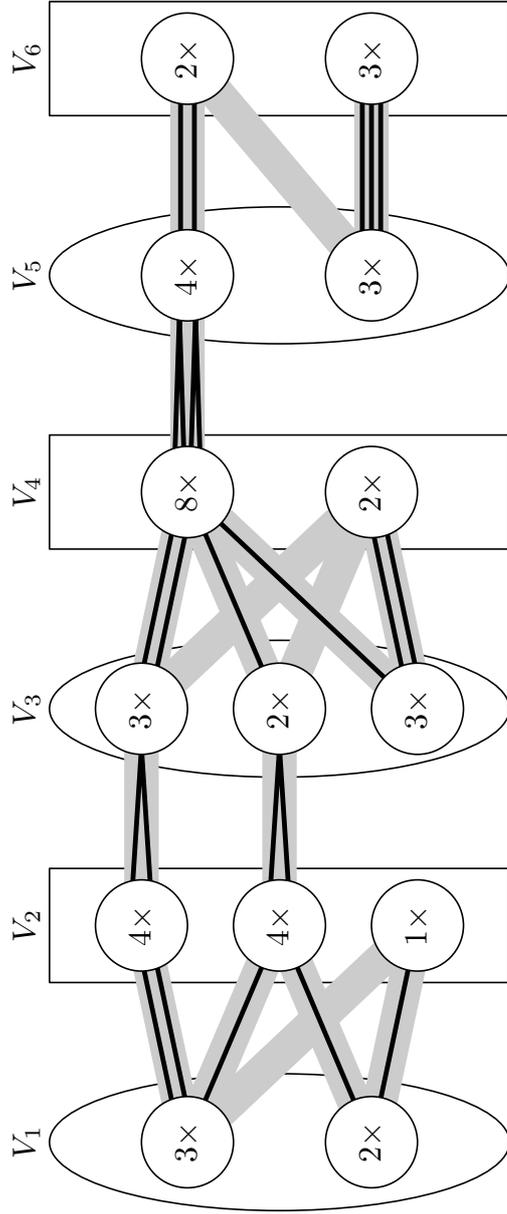


Figure 7.1: An example input B , the clusters are depicted by circles together with their sizes. The odd levels are drawn in circles and the even ones in rectangles. The edges of B are depicted by gray lines between clusters representing complete bipartite graphs. One spanning subgraph B' solving the IV-MATCHING problem is depicted in bold.

Chapter 7. Concluding Remarks

The problem IV-MATCHING asks whether there is a *spanning subgraph* $B' = (V, E')$ with each component of connectivity equal to a path of length one or two. We call this subgraph B' an IV-subgraph of B . Each vertex of an odd level V_{2t+1} is in B' adjacent either to exactly one vertex of V_{2t+2} , or to exactly two vertices of V_{2t} . Each vertex of an even level V_{2t} is adjacent to exactly one vertex of the levels $V_{2t-1} \cup V_{2t+1}$. In other words, from V_{2t-1} to V_{2t} the edges of E' form a matching, not necessarily perfect. From V_{2t} to V_{2t+1} , the edges of E' form independent V-shapes, with their centers in the level V_{2t+1} . Figure 7.1 shows an example.

Problem 7.3. *What is the complexity of the IV-MATCHING problem?*

It would be interesting to know solutions even to more restricted versions of this problem. For instance, what if the number of levels ℓ is small? We note that for instances having odd numbers of levels, as the one in Figure 7.1, we can match the highest level greedily; thus we can assume that the number of levels is always even. For two levels, the problem is just the standard perfect matching problem for bipartite graphs. On the other hand, already the case of four levels is very interesting and open. The instances of our algorithm needs to solve for the problem REGULARCOVER have at most logarithmic number of levels in size of the smaller input graph H .

Bibliography

- [1] S. B. Akers and B. Krishnamurthy. On group graphs and their fault tolerance. *IEEE Trans. Comput.*, 36(7):885–888, 1987.
- [2] D. Angluin. Local and global properties in networks of processors. In *ACM Symposium on Theory of Computing*, pages 82–93. ACM, 1980.
- [3] D. Angluin and A. Gardiner. Finite common coverings of pairs of regular graphs. *J. Combin. Theory Ser. B*, 30(2):184–187, 1981.
- [4] L. Babai. Automorphism groups of planar graphs II. In *Infinite and finite sets (Proc. Conf. Keszthely, Hungary, 1973) Bolyai-North-Holland*, pages 29–84, 1975.
- [5] L. Babai. Automorphism groups, isomorphism, reconstruction. In *Handbook of combinatorics (vol. 2)*, pages 1447–1540. MIT Press, 1996.
- [6] R. Bar-Yehuda and T. Etzion. Connections between two cycles – a new design of dense processor interconnection networks. *Discrete Applied Mathematics*, 37–38:29–43, 1992.
- [7] N. Biggs. *Algebraic graph theory*. Cambridge University Press, 1993.
- [8] O. Bílka, J. Jirásek, P. Klavík, M. Tancer, and J. Volec. On the complexity of planar covering of small graphs. In *LNCS, WG*, volume 6986, pages 83–94, 2011.

- [9] H. L. Bodlaender. The classification of coverings of processor networks. *Journal of Parallel and Distributed Computing*, 6(1):166–182, 1989.
- [10] L. Campbell. Dense group networks. *Discrete Applied Mathematics*, 37–38:65–71, 1992.
- [11] L. Campbell, G. E. Carlsson, M. J. Dinneen, V. Faber, M. R. Fellows, M. A. Langston, J. W. Moore, A. P. Mullhaupt, and H. B. Sexton. Small diameter symmetric networks from linear groups. *IEEE Trans. Comput.*, 41(2):218–220, 1992.
- [12] G. E. Carlsson, J. E. Cruthirds, H. B. Sexton, and C. G. Wright. Interconnection networks based on a generalization of cube-connected cycles. *IEEE Trans. Comput.*, 100(8):769–772, 1985.
- [13] A. Cayley. The theory of groups: Graphical representation. *Amer. J. Math.*, 1:174–176, 1878.
- [14] H. S. M. Coxeter. *Introduction to geometry*. New York, London, 1961.
- [15] P. Erdős, S. Fajtlowicz, and A. J. Hoffman. Maximum degree in graphs of diameter 2. *Networks*, 10(1):87–90, 1980.
- [16] S. Evdokimov and I. Ponomarenko. Circulant graphs: recognizing and isomorphism testing in polynomial time. *St. Petersburg Mathematical Journal*, 15(6):813–835, 2004.
- [17] H. M. Farkas and I. Kra. Riemann surfaces. In *Riemann Surfaces*, volume 71 of *Graduate Texts in Mathematics*, pages 9–31. 1992.
- [18] T. Feder and M. Y. Vardi. The computational structure of monotone monadic snp and constraint satisfaction: A study through datalog and group theory. *SIAM Journal on Computing*, 28(1):57–104, 1998.
- [19] M. R. Fellows and J. C. Stillweil. On the complexity and combinatorics of covering finite complexes. *Australasian Journal of Combinatorics*, 4:103–112, 1991.

- [20] J. Fiala. Note on the computational complexity of covering regular graphs. In *9th Annual Conference of Doctoral Students, WDS'00*, pages 89–90. Matfyzpress, 2000.
- [21] J. Fiala and J. Kratochvíl. Locally constrained graph homomorphisms structure, complexity, and applications. *Computer Science Review*, 2(2):97–111, 2008.
- [22] D. M. Goldschmidt. Automorphisms of trivalent graphs. *Annals of Mathematics*, 111(2):377–406, 1980.
- [23] J. L. Gross and T. W. Tucker. *Topological graph theory*. Courier Dover Publications, 2001.
- [24] D. Guo, J. Wu, H. Chen, and X. Luo. Moore: An extendable peer-to-peer network based on incomplete Kautz digraph with constant degree. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 821–829, 2007.
- [25] Z. Hedrlín and A. Pultr. On full embeddings of categories of algebras. *Illinois Journal of Mathematics*, 10(3):392–406, 1966.
- [26] P. Hell and J. Nešetřil. On the complexity of H -coloring. *J. Combin. Theory Ser. B*, 48(1):92–110, 1990.
- [27] A. J. Hoffman and R. R. Singleton. On moore graphs with diameters 2 and 3. *IBM Journal of Research and Development*, 4(5):497–504, 1960.
- [28] J. E. Hopcroft and R. E. Tarjan. Isomorphism of planar graphs. In *Complexity of computer computations*, pages 131–152. Springer, 1972.
- [29] J. E. Hopcroft and R. E. Tarjan. Dividing a graph into triconnected components. *SIAM Journal on Computing*, 2(3):135–158, 1973.
- [30] M. O. Katanaev. All universal coverings of two-dimensional gravity with torsion. *Journal of mathematical physics*, 34:700–736, 1993.
- [31] J. Kratochvíl, A. Proskurowski, and J. A. Telle. Covering regular graphs. *J. Comb. Theory Ser. B*, 71(1):1–16, 1997.

- [32] D. Lichtenstein. Isomorphism for graphs embeddable on the projective plane. In *ACM Symposium on Theory of Computing*, STOC '80, pages 218–224, 1980.
- [33] A. Lubiw. Some NP-complete problems similar to graph isomorphism. *SIAM Journal on Computing*, 10(1):11–21, 1981.
- [34] Rudolf Mathon. A note on the graph isomorphism counting problem. *Information Processing Letters*, 8(3):131–132, 1979.
- [35] B. D. McKay, M. Miller, and J. Širáň. A note on large graphs of diameter two and given maximum degree. *J. Combin. Theory Ser. B*, 74(1):110–118, 1998.
- [36] M. Miller and J. Širáň. Moore graphs and beyond: A survey of the degree/diameter problem. *Electronic Journal of Combinatorics*, 61:1–63, 2005.
- [37] S. Negami. The spherical genus and virtually planar graphs. *Discrete Mathematics*, 70(2):159–168, 1988.
- [38] G. Ringel and J. Youngs. Solution of the Heawood map-coloring problem. *Proceedings of the National Academy of Sciences of the United States of America*, 60(2):438–445, 1968.
- [39] J.J. Rotman. *An Introduction to the Theory of Groups*. Graduate Texts in Mathematics. Springer, 1994.
- [40] J. Stillwell. *Geometry of surfaces*. Springer, 1992.
- [41] H. Whitney. Nonseparable and planar graphs. *Trans. Amer. Math. Soc.*, 34:339–362, 1932.
- [42] S. Zhou. A class of arc-transitive Cayley graphs as models for interconnection networks. *SIAM Journal on Discrete Mathematics*, 23(2):694–714, 2009.