ON THE COMPUTATION AND COMPOSITION OF BELYĬ MAPS AND DESSINS D'ENFANTS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Jacob A. Bond

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2018

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Edray Herber Goins, Chair

Department of Mathematics

Dr. Donu V. B. Arapura

Department of Mathematics

Dr. David Ben McReynolds

Department of Mathematics

Dr. Samuel S. Wagstaff, Jr.

Department of Computer Science

Approved by:

Dr. David Goldberg

Head of the Graduate Program

For Christy, Ethan, Cecilia, and Isaac

ACKNOWLEDGMENTS

First and foremost, I am immensely grateful to my wife, Christy, for all of her love and support. Without her, I would not be where I am today. Her understanding and constant encouragement were invaluable in my pursuit of this degree. I am deeply appreciative of my children, Ethan, Cecilia, and Isaac, for all of the joy they have brought me. Their ability to brighten my day allowed me to persevere thought the most difficult parts of this dissertation. I have also benefited greatly from a caring mother who has done so much for me throughout my life.

I have been very fortunate to learn from a number of great teachers and mentors. In particular, I would like to express my sincerest gratitude to Edray Goins for introducing me to the subject of dessins d'enfants and for all of his guidance during my research and writing; to Warren Sinnott for his guidance thoughout my master's degree and for his introduction to algebraic number theory; to Jim Cogdell for his additional support during the writing of my master's thesis; to Jeff McNeal for helping me overcome my fear of, and find enjoyment in, complex analysis; to Ron Solomon for helping develop a deep understanding of abstract algebra; to David Bressoud for his priceless guidance and mentorship, as well as his excellent introduction to number theory course; to Tom Halverson for a wonderful course in representation theory which found use throughout my graduate studies; and to Dave Larabee for all of his encouragement and for the book report assignment on Fermat's Last Theorem.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

SYMBOLS

$:=$	Defined as
$\mathrm{Hom}(A, B)$	The set of morphisms from A to B
gf	The composition of g with f
$\lvert p\rvert$	The trajectory $p([0,1])$ of a path
$p_1 * p_2$	The path product $\begin{cases} p_1(2t) & \text{if } 0 \leq t \leq 1/2, \\ p_2(2t-1) & \text{if } 1/2 \leq t \leq 1. \end{cases}$
x^g	The action of g on x
$x \cdot g$	The action of g on x
$\mathrm{Aut}(X)$	The group of automorphisms of X
$\mathbb{P}^1(\mathbb{C})$	The complex points of $\mathbb{P}^1$ identified with S^2
$\mathbb{P}^1(\mathbb{R})$	The real points of $\mathbb{P}^1$ embedding $\mathbb{P}^1(\mathbb{C})$
$\mathbb{E}$	The unit disc embedded in $\mathbb{P}^1(\mathbb{C})$
$\mathbb{H}^+$	The upper half-plane embedded in $\mathbb{P}^1(\mathbb{C})$
$\mathbb{H}^-$	The lower half-plane embedded in $\mathbb{P}^1(\mathbb{C})$
$\mathbb{I}$	The unit interval $[0,1] \subseteq \mathbb{R}$; Also $\mathbb{I}$ embedded in $\mathbb{P}^1(\mathbb{C})$
$f\vert_X$	The restriction of a function f to a set X
$\pi_1(X, x)$	The fundamental group of X with base point x
Δ	A 3-constellation
$\mathrm{Mon}\, f$	The monodromy group of a covering f
id_X	The identitiy map on X: $\mathrm{id}_X(x) = x$
K^{a}	An Algebraic closure of K
$\mathrm{Gal}(L/K)$	The Galois group of L over K
$\mathrm{supp}(T)$	The support of the action of $T \leq G$ on a G-set
$H\backslash G/K$	The double cosets of H and K in G

HgK	The double coset of H and K in G with representative g
$H \rtimes_\varphi K$	The semidirect product of H and K with respect to φ
$\mathrm{Fun}(X, Y)$	The group of functions from X to Y under pointwise multiplication
$K \wr_X H$	The wreath product of K and H over the set X
$\mathcal{C}$	A conjugacy class of a group
$c_{i,j}^k$	The class multiplication coefficient of $\mathcal{C}_i$, $\mathcal{C}_j$, and $\mathcal{C}_k$
$h(\cdot)$	A cryptographic hash function
$\sim$	An equivalence relation
$\mathbb{P}^r$	Projective r-space
$\mathcal{P}_n$	The partitions of n
$C_G(g)$	The centralizer of g in G
φ	The Euler phi function
$\mathcal{O}$	An orbit of a group action
$x \vdash y$	x is a partition of y
$\langle g \rangle$	The group generated by g
$\approx$	Isomorphism (in any category)
φ_2	The Schemmel phi function
$\mathrm{mult}_p(f)$	The multiplicity of the function f at the point p
$\overline{X}$	The topological closure of X: $\bigcap_{\substack{Y \supseteq X \\ Y \text{closed}}} Y$
X°	The interior of X: $\bigcup_{\substack{Y \subseteq X \\ Y \text{open}}} Y$
∂X	The boundary of X: $\overline{X} \backslash X^\circ$
$\simeq_p$	Path homotopic
$\amalg$	Disjoint union
$p^{\circlearrowleft}$	The loop formed from a path p
$\overline{z}$	The complex conjugate of z
f_*	The postcomposition functor induced by f: $f_*(g) := f \circ g$
f^*	The precomposition functor induced by f: $f^*(g) := g \circ f$

proj_β	Projection from $G \rtimes \operatorname{Mon}\beta$ onto $\operatorname{Mon}\beta$
$\bar{\cdot}$	The image of $\cdot$ under passage to a quotient

ABSTRACT

Author: Bond, Jacob A. Ph.D.
Institution: Purdue University
Degree Received: May 2018
Title: On the Computation and Composition of BelyĬ Maps and Dessins d'Enfants
Major Professor: Edray Herber Goins.

This dissertation centers on computing with dessins d'enfants, in the form of constellations, and on the monodromy group of compositions of Belyĭ maps.

To begin, a discussion of known effective and efficient algorithms for computing constellations, Belyĭ Maps, and dessins d'enfants from one another is presented. Following this is an analysis of how to use double cosets in an optimal way to count equivalence classes of constellations. In addition, class multiplication coefficients are used to count trees with certain passports, culminating in a new proof of a result of Mednykh.

The method given by Wood for computing the constellation of a composition of Belyĭ maps is further developed and extended to allow Belyĭ maps which are defined over the complex numbers. By utilizing the fact that the monodromy group of the composition of Belyĭ maps $\beta \circ \gamma$ is a subgroup of a wreath product, generators of the monodromy group of $\beta \circ \gamma$ are found by a simple algorithm. Additionally, a group is determined from β alone which allows one to find the monodromy group of $\beta \circ \gamma$, for any γ, simply by applying the monodromy representation of γ.

Finally, using the previous results, a cryptographic protocol utilizing compositions of Belyĭ maps is proposed. A probabilistic method for efficiently deciding if the monodromy group of a Belyĭ map is either the alternating or symmetric group is discussed. Although the protocol, in its current form, is not efficient enough for

practical use, it demonstrates the ability to design a cryptographic protocol around the problem of computing Belyĭ maps.

CHAPTER 1. INTRODUCTION

1. Notation and Conventions

Composition. Let A, B, C be objects in a category. The sets of morphisms are equipped with an operation of composition [52]

$$\begin{aligned} \operatorname{Hom}(A,B) \times \operatorname{Hom}(B,C) &\to \operatorname{Hom}(A,C), \\ (f,g) &\mapsto gf. \end{aligned}$$

obtained by first applying f, then applying g.

Paths. Given a path $p : [0,1] \to X$, by abuse of notation, p will be used to denote both the function $p(t)$, as well as its image set $p([0,1])$.

Let $p_i : [0,1] \to X$, $i = 1,2$, be paths with $p_1(1) = p_2(0)$. The path which first traverses p_1, then traverses p_2, will be denoted $p_1 * p_2$.

Group Actions. Group actions will be written on the right, as is the convention in computer algebra systems [38, pg. 7], in particular in GAP, hence in Sage. As a result, permutations are multiplied from left to right. This notation is additionally forced by the monodromy representation (see Section 1.2.1.), as paths are also multiplied from left to right.

Group actions will typically be denoted by exponentiation, x^g. However, in the case that the expression of the group element is complicated, group action will be denoed by a $\cdot$, such as $x \cdot f^{g^{-1}}(y)$.

In the case of conjugation, this action is $x^g = g^{-1}xg$ [38, pg. 7]. Further, the semidirect product of groups H and K, with an action of k on h denoted by h^k, is defined to be the set $H \times K$ with group operation [19, pg. 44]

$$(u,x)(v,y) := (uv^{x^{-1}}, xy).$$

Subsets of $\mathbb{P}^1(\mathbb{C})$. Define $\mathbb{P}^1(\mathbb{C})$ to be the one-point compactification of $\mathbb{C}$, which is identified with $S^2 \subseteq \mathbb{R}^3$ as follows ($\varphi : \infty \mapsto (-1, 0, 0)$):

$$\begin{aligned}\varphi : x + iy &\longmapsto \Big(\frac{1 - x^2 - y^2}{1 + x^2 + y^2}, \frac{2x}{1 + x^2 + y^2}, \frac{2y}{1 + x^2 + y^2}\Big),\\ \frac{v + iw}{1 + u} &\longleftarrow (u, v, w).\end{aligned}$$

This has the result of sending $0, \infty, \pm i$ to $(1, 0, 0), (-1, 0, 0), (0, 0, \pm 1)$, respectively.

The following subsets of $\mathbb{P}^1(\mathbb{C})$ will be distinguished:

$$\begin{aligned}\mathbb{P}^1(\mathbb{R}) &:= \varphi(\{z \in \mathbb{C} \mid \operatorname{Im} z = 0\}) \cup \{(-1, 0, 0)\} && (= \{(u, v, w) \in S^2 \mid w = 0\}),\\ \mathbb{E} &:= \varphi(\{z \in \mathbb{C} \mid |z| < 1\}) && (= \{(u, v, w) \in S^2 \mid u > 0\}),\\ \mathbb{H}^+ &:= \varphi(\{z \in \mathbb{C} \mid \operatorname{Im} z > 0\}) && (= \{(u, v, w) \in S^2 \mid w > 0\}),\\ \mathbb{H}^- &:= \varphi(\{z \in \mathbb{C} \mid \operatorname{Im} z < 0\}) && (= \{(u, v, w) \in S^2 \mid w < 0\}).\end{aligned}$$

Additionally, let $\mathbb{I} := [0, 1] \subseteq \mathbb{R}$. By abuse of notation, let $\mathbb{I}$ also denote $\varphi(\mathbb{I} \subseteq \mathbb{C}) = \{(u, v, w) \mid 0 \leq v \leq 1, w = 0\}$.

2. Covering Spaces

Definition. *[51] Let $f : \widetilde{X} \to X$ be a continuous map. An open subset $U \subseteq X$ is evenly covered by f if $f^{-1}(U)$ is a disjoint union of open subsets S_i, called sheets, such that $f|_{S_i} : S_i \to U$ is a homeomorphism for each i.*

Definition. *[58] A continuous map $f : \widetilde{X} \to X$ is a covering map if for every $x \in X$, there is an open neighborhood of x which is evenly covered by f.*

Definition. *A lifting of a continuous function $g : Y \to X$ by a covering map $f : \widetilde{X} \to X$ is a continuous function $\tilde{g} : Y \to \widetilde{X}$ satisfying $f\tilde{g} = g$.*

Here are collected some lemmas about covering spaces which will be used in Section 4.1..

Lemma 1.1. *[48, Thm. 53.2] Let $f : \widetilde{X} \to X$ be a covering map. If $Y \subseteq X$, then $f|_{f^{-1}(Y)} : f^{-1}(Y) \to Y$ is a covering map.*

Lemma 1.2. *[58, Thm. 2.1.20] If $f : \widetilde{X} \to X$ is a covering map onto a locally path connected base space, then for any path component $\widetilde{A}$ of $\widetilde{X}$, the map $f|_{\widetilde{A}} : \widetilde{A} \to f(\widetilde{A})$ is a covering map onto some path component of X.*

Lemma 1.3. *[11, Prop. 13.3] Let $f : \widetilde{X} \to X$ be a covering map. If X is simply connected and $\widetilde{X}$ is path connected, then f is a homeomorphism.*

Lemma 1.4. *Let $f : \widetilde{X} \to X$ be a covering map. For any path connected $\widetilde{A} \subseteq \widetilde{X}$, if $f(\widetilde{A})$ is path connected, locally path connected, and simply connected, then $f|_{\widetilde{A}}$ is a homeomorphism.*

Proof. Let $A := f(\widetilde{A})$. Lemma 1.1 shows that $f|_{f^{-1}(A)}$ is a covering map. Consider the path component C of $f^{-1}(A)$ which contains $\widetilde{A}$. By Lemma 1.2, $f|_C$ is a covering map onto a path component of A. But A is path connected, so that $f|_C : C \to A$ is a covering map. Lemma 1.3 then ensures that $f|_C$ is a homeomorphism, hence $f|_{\widetilde{A}}$ is a homeomorphism. □

2.1. Monodromy Representation

Let $f : \widetilde{X} \to X$ be a covering map. Let $x \in X$ and consider $\pi_1(X, x)$. Given a loop $p \in \pi_1(X, x)$ and a point $\tilde{x} \in f^{-1}(x)$, the lifting lemma [51, Thm. 10.4] guarantees a lifting $\tilde{p}$ with $\tilde{p}(0) = \tilde{x}$. Further, $f\tilde{p}(1) = p(1) = x$ shows that $\tilde{p}(1) \in f^{-1}(x)$. In this way, setting $\tilde{x}^p = \tilde{p}(1)$, $\pi_1(X, x)$ acts transitively on $f^{-1}(x)$ [51, Thm. 10.9].

Definition. *The map $\rho : \pi_1(X, x) \to S_{p^{-1}(x)}$ defined above is the monodromy representation of f and its image in S_n is the monodromy group of f.*

If $F : X \to Y$ is a nonconstant holomorphic map of Riemann surfaces, let $\{y_1, \ldots, y_k\}$ be the branch points of F and let $X' := X \backslash \bigcup_i F^{-1}(y_i)$. Then

$$F|_{X'} : X' \to Y \backslash \{y_1, \ldots, y_k\}$$

is a covering map and the monodromy representation of F is the monodromy representation of the covering map $F|_{X'}$.

3. Objects of Study

Although the end goal is to understand dessins d'enfants, it is easier to work with the manifestation of a dessin as a constellation or a Belyĭ map, owing to their concreteness. For this reason, constellations and Belyĭ maps will be the primary objects of study.

3.1. Constellations

Definition. *A k-constellation is a sequence of k permutations $\{\sigma_i\}_{i=1}^k$, $\sigma_i \in S_n$, such that $\langle \sigma_1, \dots, \sigma_k \rangle$ is transitive on $\{1, \dots, n\}$ and*

$$\sigma_1 \cdots \sigma_k = 1.$$

Definition. *The cartographic group of a k-constellation $\{\sigma_i\}_i^k$ is the group $\langle \sigma_1, \dots, \sigma_k \rangle$. In the context of coverings of $\mathbb{P}^1(\mathbb{C})$, the cartographic group is also referred to as the monodromy group [38].*

Definition. *A passport is a k-tuple of partitions of n for some $n \in \mathbb{Z}_+$. The passport of a k-constellation is the sequence $\{t_i\}_{i=1}^k$, where t_i is the cycle type of σ_i. The set of k-constellations with a given passport forms a combinatorial orbit.*

Partitions will be written using exponents to denote multiple parts of the same size and parts will be concatenated, with either an exponent or a . separating each part. Thus, the partition $[5, 4, 3, 3, 2, 2, 2, 1]$ will be written as $5.4.3^2 2^3.1$.

There are two variations on the concept of a passport which are sometimes useful. First, when considering constellations with monodromy group $G \neq S_n$, a cycle type may not uniquely determine a conjugacy class, since conjugacy classes of S_n can split when passing to a subgroup. Additionally, because a $k-1$-constellation is determined by any $k-1$ permutations, the k-constellations whose passports agree at $k-1$ entries will sometimes be considered.

Definition. *A refined passport of a k-constellation is a k-tuple of conjugacy classes of a transitive group G. A partial passport of a k-constellation is a sequence of $k-1$ partitions of n, for some $n \in \mathbb{Z}_+$*

Definition. *Two k-constellations $\{\sigma_i\}_i$ and $\{\tau_i\}_i$ are isomorphic, denoted $\{\sigma_i\}_i \approx \{\tau_i\}_i$, if $\tau_i = \sigma_i^g$ for some $g \in S_n$ not depending on i.*

Note that both the monodromy group and the passport of a k-constellation are invariant under simultaneous conjugation of the k-constellation, hence are invariants of the isomorphism class of the k-constellation.

As the only k-constellations which will be considered are the 3-constellations, the term constellation will be taken to refer to a 3-constellation.

Having stated the primary definitions in terms of k-constellations, the case which will be studied is that of 3-constellations. For this reason, the term constellation will be taken to mean 3-constellation.

Proposition 1.5. *The constellations with passport P are in bijection with the constellations having passport P', where P' is any permutation of P.*

Proof. Consider the functions on constellations given by

$$b_1 : \begin{cases} \sigma_0 \mapsto \sigma_0' := \sigma_1 \\ \sigma_1 \mapsto \sigma_1' := \sigma_0^{\sigma_1} \end{cases}, \qquad b_2 : \begin{cases} \sigma_1 \mapsto \sigma_1' := \sigma_\infty \\ \sigma_\infty \mapsto \sigma_\infty' := \sigma_1^{\sigma_\infty} \end{cases}.$$

In each case, the third permutation of the image is defined to be the inverse of the product of the other two. If $(\sigma_0, \sigma_1, \sigma_\infty)$ has passport (t_0, t_1, t_∞), $b_1(\sigma_0)b_1(\sigma_1) = \sigma_0\sigma_1$ implies $b_1(\sigma_\infty) = \sigma_\infty$ and $b_1(\sigma_0, \sigma_1, \sigma_\infty)$ has passport (t_1, t_0, t_∞). Similarly, $b_2(\sigma_0, \sigma_1, \sigma_\infty)$ has passport (t_0, t_∞, t_1). As (1 2) and (2 3) generate S_3, upon showing that b_1, b_2 are bijections, the proposition will follow.

It will be shown that b_1 is a bijection; an analogous proof holds for b_2. First, if $(\sigma_0, \sigma_1) \approx (\tau_0, \tau_1)$, then $(\sigma_0^g, \sigma_1^g) = (\tau_0, \tau_1)$ for some $g \in S_n$. As a result,

$$\begin{aligned} b_1(\tau_0, \tau_1) &= (\tau_1, \tau_1^{-1}\tau_0\tau_1) \\ &= (\sigma_1^g, (\sigma_1^g)^{-1}\sigma_0^g\sigma_1^g) \\ &= (\sigma_1^g, (\sigma_1^{-1})^g\sigma_0^g\sigma_1^g) \\ &= (\sigma_1^g, (\sigma_0^{\sigma_1})^g) \\ &\approx (\sigma_1, \sigma_0^{\sigma_1}) = b_1(\sigma_0, \sigma_1) \end{aligned}$$

and b_1 is well-defined. Additionally, b_1 is injective, since if $b_1(\sigma_0, \sigma_1) \approx b_1(\tau_0, \tau_1)$, then there exists $g \in S_n$ so that $(\sigma_0'^g, \sigma_1'^g) = (\tau_0', \tau_1')$. But then, for $h := \sigma_1 g \tau_1^{-1}$,

$$\begin{aligned} \sigma_0^h &= \sigma_0^{\sigma_1 g \tau_1^{-1}} = (\sigma_0^{\sigma_1})^{g\tau_1^{-1}} = (\tau_0^{\tau_1})^{\tau_1^{-1}} = \tau_0, \\ \sigma_1^h &= \sigma_1^{\sigma_1 g \tau_1^{-1}} = \sigma_1^{g\tau_1^{-1}} = \tau_1^{\tau_1^{-1}} = \tau_1. \end{aligned}$$

Thus, $(\sigma_0, \sigma_1) \approx (\tau_0, \tau_1)$. Finally, the number of constellations with passports P and P' are the same by Lemma 2.2 (see Section 2.1.2.), so that b_1 is an injection of sets of equal cardinality, hence a bijection. $\square$

Definition. *Two 3-constellations $\{\sigma\}_i$ and $\{\tau_i\}_i$ are equivalent if there is a $g \in S_3$ so that $\{\sigma_{i \cdot g}\}_i \approx \{\tau_i\}_i$. Two passports are equivalent if they are permutations of each other.*

In [38], isomorphism of constellations is referred to as rigid equivalence, while equivalence is referred to as flexible equivalence.

Occasionally, it is desirable to have a canonical representative from each equivalence class of passports. When this is the case, each entry of a passport will be sorted in decreasing order, as is standard for a partition. Then the passport will be sorted in increasing lexicographic order. Justification for this convention will be given in Section 2.1.2., though the purpose is to attempt to place the partition whose conjugacy class in S_n is largest in the last position of the passport. Although sorting in this way does not guarantee the largest conjugacy class of S_n will occur last, it does hold most of the time.

Definition. *The degree of a passport is the number $n \in \mathbb{Z}_+$ for which each entry is a partition.*
The degree of a constellation is the least $n \in \mathbb{Z}_+$ so that each permutation is an element of S_n.

The following definition is a consequence of Euler's formula (see Section 1.3.3.).

Definition. *The genus g of a constellation $\Delta := (\sigma_0, \sigma_1, \sigma_\infty)$ is defined by*

$$-\deg \Delta + \sum_{i \in \{0,1,\infty\}} (\#\text{ of cycles of } \sigma_i) = 2 - 2g.$$

The genus g of a passport $P := [t_0, t_1, t_\infty]$ is defined by

$$-\deg P + \sum_{i \in \{0,1,\infty\}} (\#\text{ of parts of } t_i) = 2 - 2g.$$

3.2. Belyĭ Maps

Definition. *[38] A Belyĭ map on a Riemann surface X is a meromorphic function $\beta : X \to \mathbb{P}^1(\mathbb{C})$ which is unbranched outside of $\{0, 1, \infty\}$. In this case, (X, β) is a Belyĭ pair.*

That a constellation determines a Belyĭ map is a consequence of Riemann's existence theorem, choosing $(y_1, y_2, y_3) = (0, 1, \infty)$.

Riemann's existence theorem. *[38] Fix points $\{y_1, \ldots, y_k\}$. For any k-constellation $[g_1, \ldots, g_k]$, $g_i \in S_n$, there exists a compact Riemann surface X and a meromorphic function $f : X \to \mathbb{P}^1(\mathbb{C})$ such that $y_1, \ldots, y_k$ are the critical values of f, and $g_1, \ldots, g_k$ are the corresponding monodromy permutations.*

Just as for constellations, there are notions of isomorphism and equivalence for Belyĭ maps.

Definition. *Let β_1, β_2 be Belyĭ maps with domain X. Let μ_1 be an automorphism of X and μ_2 be an automorphism of $\mathbb{P}^1(\mathbb{C})$ so that $\mu_2(\{0, 1, \infty\}) = \{0, 1, \infty\}$. Then β_1 and β_2 are isomorphic if $\beta_1 = \beta_2\mu_1$. Additionally, β_1 and β_2 are equivalent if $\beta_1 = \mu_2\beta_2\mu_1$.*

In this way, Belyĭ maps reflect the rearrangment of a passport through the ability to postcompose with μ_2.

Definition. *The degree of a Belyĭ map is its degree as a map of Riemann surfaces. The genus of a Belyĭ map is the genus of its domain.*

Definition. *An edge of a Belyĭ map β is a lifting of* $\mathrm{id}_{\mathbb{I}} : x \mapsto x$ *by β.*

Lemma 1.6. *Define $\mathbb{I}^\circ$ be $\mathbb{I} \backslash \{0, 1\}$ and let e_β be an edge of β. Then $e_\beta|_{\mathbb{I}^\circ}$ is the restriction to $\mathbb{I}^\circ \subseteq \mathbb{E}$ of a holomorphic map on $\mathbb{E}$.*

Proof. Let e_β be an edge of β and let $\widetilde{A}$ be the path component of $\beta^{-1}(\mathbb{E})$ containing e_β. By Lemma 1.4, β is a homeomorphism on $\widetilde{A}$. In particular, $\beta|_{\widetilde{A}}$ is injective, and being an injective holomorphic map, $\beta|_{\widetilde{A}}$ is an isomorphism [47, Prop. 3.9] and there exists a holomorphic inverse f. As $\beta f(\mathbb{I}^\circ) = \mathbb{I}^\circ = \beta e_\beta(\mathbb{I}^\circ)$ and β is injective on $\widetilde{A}$, the result follows. □

3.2.1. Dynamical Belyĭ Maps

In order that a composition of Belyĭ maps $\beta\gamma$ be a Belyĭ map, it is necessary that β have genus zero. However, this condition is not sufficient.

Definition. *[38] A Belyĭ map $\beta : \mathbb{P}^1(\mathbb{C}) \to \mathbb{P}^1(\mathbb{C})$ is a dynamical Belyĭ map if $\beta(\{0, 1, \infty\}) \subseteq \{0, 1, \infty\}$.*

Lemma 1.7. *[38, cf. Prop. 2.5.17] If $\gamma : X \to \mathbb{P}^1(\mathbb{C})$ is a Belyĭ map and β is a dynamical Belyĭ map, then $\beta\gamma$ is a Belyĭ map.*

3.3. Dessins d'Enfants

Definition. *[55] A Grothendieck dessin (d'enfant) is a sequence $X_0 \subset X_1 \subset X_2$ such that X_2 is a compact connected Riemann surface, $X_2 \backslash X_1$ is a finite disjoint union of connected open sets, each homoemorphic to $\mathbb{E}$, $X_1 \backslash X_0$ is a finite disjoint union of curves, and X_0 is a finite set of points with a fixed bipartite structure.*

Definition. *An (abstract) dessin (d'enfant) is a graph with labeled edges and endowed with a fixed bipartite structure for which there is a cyclic ordering of the edges around each vertex.*

A dessin is recovered from a Grothendieck dessin by forgetting its complex structure, retaining only the orientation of the curves of $X_1 \backslash X_0$ around the points of X_0.

On the other hand, the cyclic ordering of edges of a dessin can be used to recover an associated Grothendick dessin [8, pg. 28].

A Grothendieck dessin can be obtained from a Belyĭ map as $\beta^{-1}(\mathbb{I})$. The reason for using $\mathbb{I}$ is that the segments $[0, 1/2]$ and $[1/2, 1]$ connect the base point $1/2$ to all but one of the branch points of β. Hence loops around the endpoints generate $\pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}, 1/2)$ and the preimages of $\beta^{-1}([0, 1])$ glue together at $\beta^{-1}(0)$ and $\beta^{-1}(1)$, so that $\beta^{-1}(\mathbb{I})$ is connected. This is discussed in the case of a polynomial in [38, pg. 85]

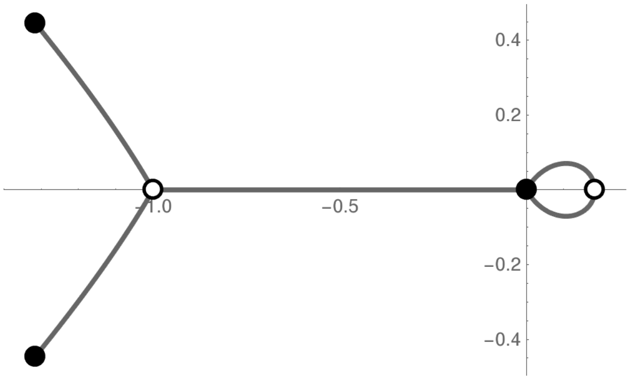


Figure 1.1.: $\beta^{-1}([0, 1])$ for $\beta(x) = \dfrac{15x^3(3x^2 + (2a + 3)x + 4a - 4)}{(4a - 9)(15x - 2a + 3)}$, where $a = \sqrt{6}$.

The cyclic ordering of the labeled edges around each vertex of a dessin creates a pair of permutations, one for the vertices in each subset of the bipartition. The third permutation of a constellation is then recovered by either reading the edge labels going around each face [8, pg. 29] or simply taking $\sigma_\infty = (\sigma_0\sigma_1)^{-1}$.

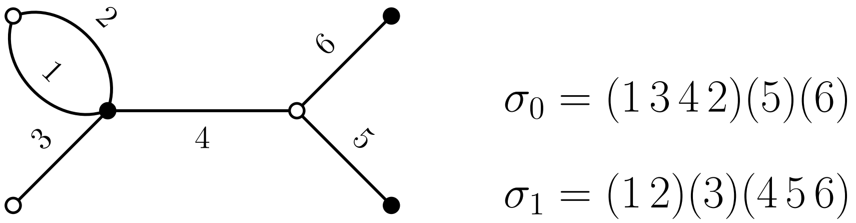

2
1
3
4
6
5
$\sigma_0 = (1\,3\,4\,2)(5)(6)$
$\sigma_1 = (1\,2)(3)(4\,5\,6)$

Figure 1.2.: A dessins d'enfant and the corresponding pair of permutations

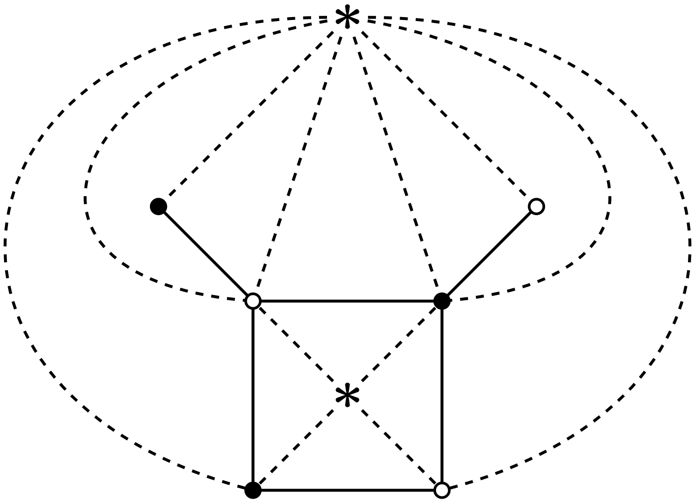


Figure 1.3.: The canonical triangulation of a dessin d'enfant

3.3.1. Canonical Triangulation

Definition. *[55] Given a Grothendieck dessin (X_0, X_1, X_2), place a point v_i in each connected component of $X_2 \backslash X_1$. Then the canonical triangulation of (X_0, X_1, X_2) is the set of triangles whose three vertices are*

1. *the two endpoints of a segment e in $X_1 \backslash X_0$,*
2. *a vertex v_i such that e is in the closure of the connected component containing v_i.*

The canonical triangulation of a dessin is obtained by an analogous procedure.

As the vertices of X_0 have a bicoloring, marking the vertices v_i with a $*$ results in the triangles of the canonical triangulation having vertices marked, one each, by $\bullet$, $\circ$, and $*$. Moreover, two types of triangles are created, depending on the orientation of the marked vertices. Note that a Riemann surface with finitely many points removed is still a Riemann surface, hence is orientable.

Definition. *If, when traversing in the counterclockwise direction the boundary of a triangle T, the vertices of T appear in the order $\bullet$, $\circ$, and $*$, then T is a positive triangle. If the vertices appear in the reverse order, T is a negative triangle [38].*

3.3.2. Equivalence and Passports

Definition. *Two Grothendieck dessins are isomorphic if there is an automorphism of the surface X_2 taking one Grothendieck dessin to the other. Two dessins are isomorphic if their canonical triangulations are isomorphic.*

Two dessins are equivalent if their canonical triangulations are isomorphic up to a recoloring of their vertices.

In particular, because relabeling the edges of a dessin, which corresponds to simultaneously conjugating the associated constellation, does not change the canonical triangulation, two dessins which differ only in the labeling of their edges are isomorphic.

Definition. *The passport for a dessin D has three elements, the first of which gives the degrees of the black vertices of D, the second gives the degrees of the white vertices of D, and the third gives the degrees of the faces of D (the degree of a face is half the degree of the vertex of the canonical triangulation which is placed in the face).*

Because the vertices of the canonical triangulation may be colored in any manner without affecting the equivalence class of the canonical triangulation, this illustrates that the passport of a dessin can rearranged without affecting the equivalence class, just as with constellations and Belyĭ maps.

Definition. *The degree of a dessin or Grothendieck dessin is the number its edges. The genus of a Grothendieck dessin is the genus of the surface in which it is embedded. The genus of a dessin is the smallest $n \in \mathbb{Z}_+$ such that it can be embedded in a surface of genus n.*

3.4. Galois Action

Belyĭ's theorem. *[5] A complete nonsingular algebraic curve X defined over a field of characteristic zero can be defined over $\mathbb{Q}^{\mathrm{a}}$ iff it can cover $\mathbb{P}^1(\mathbb{C})$ with ramification over three points.*

A self-contained proof of this theorem can be found in [34]. Belyĭ gave a second, more explicit, proof of the "only if" direction in [6].

As such, Belyĭ maps are defined over number fields so that $\mathrm{Gal}(\mathbb{Q}^{\mathrm{a}}/\mathbb{Q})$ acts on Belyĭ maps and, by extension, on their associated dessins. The primary interest in dessins is due to this action of $\mathrm{Gal}(\mathbb{Q}^{\mathrm{a}}/\mathbb{Q})$. Not only is the action faithful, it is faithful even when restricted to plane trees [55, Thm. II.4]. Another important consideration of this action is the orbits which are formed.

Definition. *Let (X, β) be a Belyĭ pair. The Galois orbit of (X, β) is the set*

$$\{(X^{\sigma}, \beta^{\sigma}) \mid \sigma \in \mathrm{Gal}(\mathbb{Q}^{\mathrm{a}}/\mathbb{Q})\}.$$

The Galois orbit of a Grothendieck dessin, dessin, constellation with Belyĭ pair (X, β) is the set of Grothendieck dessins, dessins, constellations, repsectively, whose associated Belyĭ pair is in the Galois orbit of (X, β).

Among the invariants of a Galois orbit are the passport and monodromy group of a dessin d'enfant. It is an open problem to determine a collection of invariants which can separate Galois orbits. Although the Belyĭ-extending maps introduced by Wood [68] could likely be used to separate any two given orbits, their role in separating a generic pair of orbits is less clear.

4. Groups and Representations

Definition. *[19] Let the group G act transitively on a set A. Then $B \subseteq A$ is a block if $B^g = B$ or $B^g \cap B = \emptyset$ for all $g \in G$.*

Definition. *[19] Let the group G act on a set A, and let $T \subseteq G$ be a subset of G. Then the support of T is*

$$\mathrm{supp}(T) := \{a \in A \mid a^t \neq a \textit{ for some } t \in T\}.$$

In analogy with the support of real-valued functions, considered as an additive group, the support of T is the complement of the set S of points in A with $T|_S = 1$.

Let H, K be subgroups of a group G. Then define the double coset HgK, $g \in G$, to be the set

$$HgK := \{hgk \in G \mid h \in H, k \in K\}.$$

Alternatively, the double coset HgK is the equivalence class of elements g' of G so that $g' = hgk$ for some $h \in H$ and $k \in K$. The collection of double cosets of H and K is denoted $H\backslash G/K$.

Let A be a set and let H and K be groups with H acting on A. Consider the set $\mathrm{Fun}(A, K)$ of functions from A to K. Then $\mathrm{Fun}(A, K)$ becomes a group under the operation of pointwise multiplication:

$$(f \cdot g)(a) := f(a)g(a).$$

As H acts on A, H also acts on $\mathrm{Fun}(A, K)$ by

$$f^h(a) := f(a^{h^{-1}}).$$

This gives a homomorphism $\varphi : H \to \mathrm{Aut}\Big(\mathrm{Fun}(A, K)\Big)$ and wreath product of K and H over A is

$$K \wr_A H := \mathrm{Fun}(A, K) \rtimes_\varphi H.$$

4.1. Class Multiplication Coefficients

Definition. *Let $\mathcal{C}_i, \mathcal{C}_j, \mathcal{C}_k$ be three conjugacy classes of a group G and fix $g \in \mathcal{C}_k$. Then the number of solutions to $xy = g$ with $x \in \mathcal{C}_i, y \in \mathcal{C}_j$ is the class multiplication coefficient $c_{i,j}^k$.*

The class multiplication coefficient is independent of the choice of $g \in \mathcal{C}_k$ since $g = xy$ iff $g^h = x^h y^h$. An important feature of the class multiplication coefficients is that they can be determined from the character table of G.

Theorem 1.8. *[28] Let G be a group, let $\mathcal{C}_1, \dots, \mathcal{C}_m$ be the conjugacy classes of G, and let $g_i \in \mathcal{C}_i$ for all $1 \leq i \leq m$. Then the class multiplication coefficient $c_{i,j}^k$ is given by*

$$c_{i,j}^k = \frac{|G|}{|C_G(g_i)||C_G(g_j)|} \sum_{\chi} \frac{\chi(g_i)\chi(g_j)\overline{\chi(g_k)}}{\chi(1)},$$

where the sum is over all irreducible characters χ of G.

Remark 1. Inducting on n allows one to obtain a corresponding formula for solutions to $\prod_i^n x_i = g$ with $x_i \in \mathcal{C}_i$ [56].

The class multiplication coefficients give an upper bound on the number of constellations with a given refined passport. Let $P = (\mathcal{C}_i, \mathcal{C}_j, \mathcal{C}_k)$ be a refined passport and let $g \in \mathcal{C}_k$. Let Σ be the set of solutions to $xy = g$ with $x \in \mathcal{C}_i, y \in \mathcal{C}_j$. If $\Delta := (\sigma_0, \sigma_1, \sigma_\infty)$ is a constellation with refined passport P, then Δ can be simultaneously conjugated so that $\sigma_\infty = g$. As such, Δ is isomorphic to a constellation in Σ and $c_{i,j}^k$ provides an upper bound for the number of constellations with refined passport P. However, it is possible, and even likely (see Section 3.2.2.), that Σ will contain many constellations which are isomorphic to Δ. Moreover, given a pair $(x, y) \in \Sigma$, there is no guarantee that $\langle x, y \rangle$ will be transitive, and even if it is, $\langle x, y \rangle$ may generate a proper subgroup of G.

Nevertheless, $c_{i,j}^k$ gives an upper bound on the size of the combinatorial orbit of Δ. In particular, if a class multiplication coefficient is 0, this guarantees that there are no constellations with a given refined passport. Additionally, Mednykh [45] was able to work out a formula for the number of k-constellations having a given passport using inclusion-exclusion to handle the case where the sequence of permutations generates an intransitive group, though the formula is not easily computable except in certain special cases.

5. Cryptography

A few definitions related to cryptography will be used in Chapter 5.

Definition. *A security parameter is a parameter of a cryptographic protocol which determines the minimum allowable complexity for objects involved in the protocol.*

Definition. *[32] Let $\{0,1\}^* := \bigcup_{\ell=0}^\infty \{0,1\}^\ell$, where $\{0,1\}^\ell$ is the ℓ-fold Cartesian product. A hash function is a function $h : \{0,1\}^* \to \{0,1\}^\ell$ for a fixed ℓ. A cryptographic* *hash function is a hash function with a security parameter such that for all probabilistic polynomial-time adversaries $\mathcal{A}$,*

$$\Pr\Big(\mathcal{A} \textit{ finds } x, x' \in \{0,1\}^* \textit{ with } h(x) = h(x')\Big)$$

is a negligible function of the security parameter of the hash function.

In informal terms, a cryptographic hash function is a hash function for which one has a negligible probability of finding a collision.

CHAPTER 2. COMPUTATION

The algorithms discussed in this chapter, as well as in Chapter 3 and 4, have been implemented for Sage and are available at `https://gitlab.com/jacobbond/dessins`.

1. Constellations and Passports

1.1. Constellations to Passports

This is trivial and is just the determination of the cycle types of each permutation in the constellation.

Algorithm 2.1 Finding the passport of a constellation

```
function ConstellationToPassport($\sigma_0$, $\sigma_1$, $\sigma_\infty$)
passport ← []
for $\sigma$ in [$\sigma_0$, $\sigma_1$, $\sigma_\infty$] do
    append cycle type of $\sigma$ to passport
return passport
```

1.2. Passports to Constellations

In generating the constellations with passport (t_0, t_1, t_∞), one is looking for the triples $(\sigma_0, \sigma_1, \sigma_\infty) \in \mathcal{C}_{t_0} \times \mathcal{C}_{t_1} \times \mathcal{C}_{t_\infty}$ which generate a transitive group and multiply to 1. However, only one representative from each isomorphism class of constellations should be generated. In order to reduce from a search over $\mathcal{C}_{t_0} \times \mathcal{C}_{t_1} \times \mathcal{C}_{t_\infty}$ to a search of the orbits of $\mathcal{C}_{t_0} \times \mathcal{C}_{t_1}$ under simultaneous conjugation, S_n will be quotiented by the centralizers of and element $g_i \in \mathcal{C}_{t_i}$, $i = 0, 1$, and σ_∞ will be taken to be $(\sigma_0\sigma_1)^{-1}$.

Lemma 2.2. *[33] Let G be a group and let $\mathcal{C}_0, \mathcal{C}_1$ be conjugacy classes of G with representatives τ_0, τ_1, respectively. Then*

$$\begin{aligned} C_G(\tau_0)\backslash G/C_G(\tau_1) &\longleftrightarrow \{(\sigma_0, \sigma_1) \mid \sigma_i \in \mathcal{C}_i\}\Big/ \sim_G \\ C_G(\tau_0)gC_G(\tau_1) &\longmapsto (\tau_0^g, \tau_1) \end{aligned}$$

is a bijection, where $\sim_G$ is equivalence under simultaneous conjugation by elements of G.

Proof. The map is well-defined, since if h_0gh_1 is another representative of $C_G(\tau_0)gC_G(\tau_1)$, then $h_i \in C_G(\tau_i)$, and

$$\begin{aligned} h_0gh_1 &\mapsto (\tau_0^{h_0gh_1}, \tau_1) \\ &\sim_G (\tau_0^{gh_1h_1^{-1}}, \tau_1^{h_1^{-1}}) \\ &= (\tau_0^g, \tau_1). \end{aligned}$$

For $(\sigma_0, \sigma_1) \in \mathcal{C}_0 \times \mathcal{C}_1$, there exist h_0, h_1 so that $\sigma_i = \tau_i^{h_i}$. Then

$$h_0h_1^{-1} \mapsto (\tau_0^{h_0h_1^{-1}}, \tau_1) \sim_G (\sigma_0, \tau_1^{h_1}) = (\sigma_0, \sigma_1)$$

and the map is surjective. Finally, if $g' \mapsto (\tau_0^g, \tau_1)$, then there exists h_1 so that

$$(\tau_0^{g'h_1}, \tau_1^{h_1}) \sim_G (\tau_0^g, \tau_1),$$

implying $h_1 \in C_G(\tau_1)$. Setting $h_0 := g'h_1g^{-1}$,

$$\tau_0^{h_0} = \tau_0^{g'h_1g^{-1}} = \tau_0.$$

Then $h_0 \in C_G(\tau_0)$, $g' = h_0gh_1^{-1}$, and the map is injective. □

This lemma allows for at least a few different approaches to the problem of generating constellations, though in all cases, it must be checked that all pairs are transitive on the moved points of G. If one is interested in all constellations with passport P, G should be taken to be S_n. This is the approach that will be followed here. Alternatively, one can start with a transitive group $G \neq S_n$, in which case, only pairs of permutations which generate subgroups of G will be recovered. If one is looking specifically for constellations with monodromy group G, it must be checked whether the resulting pairs generate the whole group G. When $G \neq S_n$, a cycle type does not necessarily determine a unique conjugacy class of G, so that in this case, a refined passport is the appropriate input.

Algorithm 2.3 Finding the constellations with passport $(t_0, t_1, \cdot)$

```
function PartialPassportToConstellations($t_0$, $t_1$)
deg ← sum($t_0$)
$G$ ← SymmetricGroup(deg)
$\tau_0$,$\tau_1$ ← a permutation with cycle type $t_0$,$t_1$, respectively
$C_0$,$C_1$ ← $C_G(\tau_0)$,$C_G(\tau_1)$
coset_reps ← DoubleCosetRepresentatives($G$,$C_0$,$C_1$)
constellations ← []
for $g$ in cosetreps do
    $\tau_0' \leftarrow \tau_0^g$
    if ($\tau_0'$,$\tau_1$) is transitive on $\{1, \ldots,$ deg$\}$ then
        append ($\tau_0'$,$\tau_1$) to constellations
return constellations
```

For efficiency purposes, in the above algorithm, it is preferable to use a function which computes representatives without actually creating the double cosets, such as GAP's `DoubleCosetRepsAndSizes`.

Note that only two conjugacy classes are taken as input to the algorithm. As such, all constellations with partial passport (t_0, t_1) will be returned, regardless of the cycle type of σ_∞. If desired, the resulting constellations can then be filtered based on the cycle type of σ_∞. However, when looking for constellations with a given passport, the third cycle type can sometimes be utilized to improve the efficiency of the procedure by generating the constellations having an equivalent passport instead (see Proposition 1.5).

When the subgroups forming the double cosets are larger, each double coset is larger, and the number of double cosets is smaller. Thus, choosing larger centralizers will result in fewer double cosets and will be computationally less expensive. As the size $|C_G(g)|$ of a centralizer is $|G|/|\mathcal{C}_g|$ by the orbit-stabilizer theorem, this corresponds to choosing smaller conjugacy classes. The following lemma can also be derived from Lemma 3.10.

Lemma 2.4. *The number of elements of S_n with cycle type $(t_1^{a_1}, \ldots, t_m^{a_m})$ is given by*

$$\frac{\left(\sum_{i=1}^m t_i\right)!}{\prod_{i=1}^m t_i^{a_i}(a_i!)}.$$

Proof. There are $(\sum t_i)!$ ways to arrange the $\sum t_i$ points. Each cycle of length t_i can be written in t_i distinct ways and a collection of a_i t_i-cycles can be written in $a_i!$ distinct ways. □

As an example of the impact that rearranging a passport can have, consider a search for constellations with passport $(13, 7.6, 5.4^2)$.

```
gap> G:=SymmetricGroup(13);;
gap> C0:=Centralizer(G, (1,2,3,4,5,6,7,8,9,10,11,12,13));;
gap> C1:=Centralizer(G, (1,2,3,4,5,6,7)(8,9,10,11,12,13));;
gap> reps_sizes:=DoubleCosetRepsAndSizes(G, C0, C1);; time;
269403
gap> # 269403 miliseconds = 5.49 minutes
gap> C0:=Centralizer(G, (1,2,3,4,5)(6,7,8,9)(10,11,12,13));;
gap> reps_sizes:=DoubleCosetRepsAndSizes(G, C0, C1);; time;
44760
gap> # 44760 miliseconds = 44.76 seconds
```

In each of the above cases, all equivalence classes of constellations having a passport equivalent to $(13, 7.6, 5.4^2)$ will be recovered. Making use of this optimization leads to Algorithm 2.5.

Algorithm 2.5 Finding the constellations with passport (t_0, t_1, t_∞)

```
function PassportToConstellations($t_0$, $t_1$, $t_\infty$)
num_conjugates ← []
for cycle_type in ($t_0$,$t_1$,$t_\infty$) do
    append NumConjugates(cycle_type) to num_conjugates
  ▷ Determine how to reorder the partitions:
if num_conjugates[2] == max(num_conjugates) then
    define Replace($s_0$,$s_1$,$s$):  return ($s_0$,$s_1$,$s$)
else if num_conjugates[1] == max(num_conjugates) then
    $t_1, t_\infty \leftarrow t_\infty, t_1$
    ▷ Algorithm 2.3 returns $(s_0, s_\infty, \cdot)$ with types $(t_0, t_\infty, t_1)$:
    define Replace($s_0$,$s_\infty$,$s$):  return ($s_0$,$(s_\infty s_0)^{-1}$,$s_\infty$)
else if num_conjugates[0] == max(num_conjugates) then
    $t_0, t_\infty \leftarrow t_\infty, t_0$
    ▷ Algorithm 2.3 returns $(s_\infty, s_1, \cdot)$ with types $(t_\infty, t_1, t_0)$:
    define Replace($s_\infty$,$s_1$,$s$):  return ($(s_1 s_\infty)^{-1}$,$s_1$,$s_\infty$)
partial_constellations ← PartialPassportToConstellations($t_0$,$t_1$)
constellations ← []
for constell in partial_constellations do
    if cycle type of constell[2] is $t_\infty$ then
        append Replace(constell) to constellations
return constellations
```

1.2.1. Notes

Van Hoeij [65] has an algorithm, the RoadMap algorithm, for computing the dessins, hence constellations (see Section 2.4.1.), with a given passport. Starting from an empty dessin, edges are inserted one at a time until the dessins with desired passport are reached. However, as many dessins of a given degree can all be extended to the same dessin having the desired passport, intermediate dessin are discarded as soon as it can be determined that they are unnecessary. In certain cases, the RoadMap algorithm can find all constellations having passports of very large degrees. As a particularly favorable example, the RoadMap algorithm was able to find the 11 constellations with passport $(2^{48}, 3^{32}, 6^{16})$.

1.3. Determining Equivalence of Constellations

Due to the equivalence relation on constellations, it can be difficult to determine whether two constellations with the same passport are equivalent. However, the ability to do so is useful in general, and will be used in particular, when determining the Belyĭ map of a constellation.

1.3.1. Checking for Simultaneous Conjugacy

To begin, Lemma 2.2 provides a simple method for detecting simultaneous conjugacy. Given $(\sigma_0, \sigma_1), (\sigma_0', \sigma_1') \in \mathcal{C}_0 \times \mathcal{C}_1$, there exist $h_0, h_1 \in G$ so that $(\sigma_0^{h_0}, \sigma_1^{h_1}) = (\sigma_0', \sigma_1')$ and $(\sigma_0^{h_0 h_1^{-1}}, \sigma_1) \sim_G (\sigma_0', \sigma_1'$. It is trivial to find such h_i by matching cycles of σ_i and σ_i' of the same length and mapping the ith element of one to the other.

Let $h := h_0 h_1^{-1}$. According to the bijection of Lemma 2.2, the simultaneous conjugacy class of (σ_0', σ_1') is the image of $C_G(\sigma_0) h C_G(\sigma_1)$. As (σ_0, σ_1) is the image of $C_G(\sigma_0) C_G(\sigma_1)$,

$$(\sigma_0, \sigma_1) \sim_G (\sigma_0', \sigma_1') \iff C_G(\sigma_0) C_G(\sigma_1) = C_G(\sigma_0) h C_G(\sigma_1) \iff h \in C_G(\sigma_0) C_G(\sigma_1).$$

However, the entirety of $C_G(\sigma_0) C_G(\sigma_1)$ need not be considered. First,

$$h \in C_G(\sigma_0) C_G(\sigma_1) \iff \exists g \in C_G(\sigma_1) \text{ with } hg^{-1} \in C_G(\sigma_0).$$

But $hg^{-1} \in C_G(\sigma_0)$ iff for all $k \in C_G(\sigma_0) \cap C_G(\sigma_1)$, $kg \in C_G(\sigma_1)$ and $hg^{-1}k^{-1} \in C_G(\sigma_0)$. As such, any $g \in C_G(\sigma_1) / \big(C_G(\sigma_0) \cap C_G(\sigma_1)\big)$ accomplishes $(\sigma_0^{hg^{-1}}, \sigma_1^{hg^{-1}}) = (\sigma_0', \sigma_1')$. Thus, if h satisfies $(\sigma_0^h, \sigma_1) \sim_G (\sigma_0', \sigma_1')$,

$$(\sigma_0, \sigma_1) \sim_G (\sigma_0', \sigma_1') \iff \exists g \in C_G(\sigma_1) / \big(C_G(\sigma_1) \cap C_G(\sigma_0)\big) \text{ with } hg^{-1} \in C_G(\sigma_0).$$

As mentioned, finding an appropriate h is trivial, so that this reduces the question of simultaneous conjugacy to iterating over $g \in C_G(\sigma_1)/\big(C_G(\sigma_1) \cap C_G(\sigma_0)\big)$ and checking whether $\sigma_0^{hg^{-1}} = \sigma_0$.

1.3.2. Finding a Canonical Representative

The disadvantage of the previous approach is that it is only a pairwise comparison. Thus, in a list of n nonisomorphic constellations, it would be necessary to call the function $\binom{n}{2}$ times to determine that none of the constellations were isomorphic. On the other hand, if one had a function to determine a canonical representative of each equivalence class of constellations, it would only be necessary to call this function n times, compute a hash of each output, and make at most $\binom{n}{2}$ comparisons of integers, rather than checking for simultaneous conjugacy. Such a function is given in [44] or [66]. Note that aside from the optimizations suggested in [44], the only difference between the two algorithms is the order in which an index set $I \times J$ is traversed.

As the computation of a canonical representative for a k-constellation is identical for all k, k-constellations will be considered for this discussion. For each $i \in \{1, \ldots, n\}$, a given k-constellation can be relabelled as follows. Let Q be a double-ended queue initially containing i and let L be a list initially containing i. While $\#L \neq n$, pop the head j of Q and, for $j^{\sigma_1}, j^{\sigma_2}, \ldots, j^{\sigma_k}$, if $j^{\sigma_\ell} \notin L$,

1. push j^{σ_ℓ} into Q at the tail,
2. append j^{σ_ℓ} to L.

Then defining $a^{\tau_i} := L[a-1]$, simultaneous conjugation by τ_i^{-1} defines a relabeling of the constellation. Having performed this relabeling for each $i \in \{1, \ldots, n\}$, choose the canonical representative to be the lexicographically least element among the relabeled permutations

$$\Big\{(\sigma_1, \ldots, \sigma_k)^{\tau_i^{-1}} \mid i \in \{1, \ldots, n\}\Big\},$$

where $(\sigma_1, \ldots, \sigma_k)^{\tau_i^{-1}}$ is simultaneous conjugation by τ_i^{-1}.

Note that τ is just the inverse of the permutation defined by `labeling`.

Algorithm 2.6 Finding a canonical representative for a constellation

```
function CanonicalConstellation($\sigma_1$, $\sigma_2$, …, $\sigma_k$)
candidates, deg ← [], max(NumberOfMovedPoints($\sigma_i$))
for $1 \leq$ i $\leq$ deg do
    append RelabelByBase(i,$(\sigma_1, \ldots, \sigma_k)$) to candidates
return Min(candidates)
```

Algorithm 2.7 Relabel a permutation starting from a given point

```
function RelabelByBase($i$, $(\sigma_1, \ldots, \sigma_k)$)
q, labeling ← deque([i]), [i]
while size(seen) < deg do
    $x$ ← PopLeft(Q)
    for $\sigma$ in $(\sigma_1, \ldots, \sigma_k)$ do
        if $x^\sigma$ not in labeling then
            append $x^\sigma$ to labeling
            PushRight(Q, $x^\sigma$)
▷ relabel $a$ by its position in labeling
define $\tau$ by $a^\tau = b$ if labeling[$b - 1$] = $a$
return $(\sigma_1, \ldots, \sigma_k)^\tau$
```

Lemma 2.8. *[66] $\Delta_1 \approx \Delta_2$ iff for $j = 1, 2$,*

$$\{\texttt{RelabelByBase(i,}\Delta_j\texttt{)} \mid 1 \leq i \leq \deg \Delta_j\}$$

coincide.

Proof. That the latter condition is sufficient is clear. To see that it is necessary, it will be shown that for $g \in S_n$,

$$\texttt{RelabelByBase(i,}\Delta\texttt{)} = \texttt{RelabelByBase(i}^g\texttt{,}\Delta^g\texttt{)}.$$

Thus, if $\Delta_1 \approx \Delta_2$, then for some $g \in S_n$,

$$\texttt{RelabelByBase(i, }\Delta_1\texttt{)} = \texttt{RelabelByBase(i}^g\texttt{, }\Delta_2\texttt{)}.$$

As g is a permutation,

$$\{\texttt{RelabelByBase(i}^g\texttt{, }\Delta_2\texttt{)}\}_{i=1}^{\deg \Delta_2} = \{\texttt{RelabelByBase(i, }\Delta_2\texttt{)}\}_{i=1}^{\deg \Delta_2}$$

and the result follows.

Consider the execution of `RelabelByBase(i, Δ)` and $\texttt{RelabelByBase(i}^g\texttt{, }\Delta^g\texttt{)}$. For each quantity $\cdot$ in the former, let $\cdot_g$ denote the corresponding quantity in the latter. Observe that during the first iteration of the **while** loop on input (i, Δ), $x = i$ and so the elements appended to `labeling` are of the form i^σ for some $\sigma \in \Delta$. On the other hand, on input (i^g, Δ^g), $x_g = i^g$ and each element appended to `labeling` has the form $(i^g) \cdot \sigma^g = i^{\sigma g}$. Moreover, $i^\sigma = i^{\sigma'}$ iff $i^{\sigma g} = i^{\sigma' g}$, so that i^σ is appended to `L` iff $i^{\sigma g}$ is appended to $\texttt{L}_g$.

By induction, in further iterations of the **while** loop, x will be of the form i^η on input (i, Δ) and $i^{\eta g}$ on input (i^g, Δ^g), where η is a product of elements of Δ. Then each appended element is of the form $i^{\eta\sigma}$ for some $\sigma \in \Delta$ in the former case, and $i^{\eta\sigma g}$ in the latter case, and $i^{\eta\sigma}$ is appended iff $i^{\eta\sigma g}$ is appended. The end result is that if $L[b-1] := a$, then $L_g[b-1] = a^g$ and $a^{g\tau_g} = b = a^\tau$. Thus

$$\texttt{RelabelByBase(i}^g\texttt{, }\Delta^g\texttt{)} = (\Delta^g)^{\tau_g} = \Delta^\tau = \texttt{RelabelByBase(i, }\Delta\texttt{)},$$

as was to be shown. □

2. Passports and Belyĭ Maps

2.1. Passports to Belyĭ Maps

The approach presented here is restricted to Belyĭ maps which have genus zero. A method using modular forms to compute a Belyĭ map from a constellation of arbitrary genus is given in [33].

2.1.1. Constructing a Polynomial System

Given a passport of genus zero, the first step will be to create a multivariate polynomial system as follows. The approach was originally described in [2], though the descriptions given in [8, 57] will be followed here. Let $P := (\prod_\ell \ell^{a_\ell}, \prod_\ell \ell^{b_\ell}, \prod_\ell \ell^{c_\ell})$ and let β be a Belyĭ map with passport P. Then for each ℓ, the Belyĭ map β

has a_ℓ roots of multiplicity ℓ, and its numerator contains the factor p_ℓ^ℓ, where p_ℓ is a polynomial of degree a_ℓ. Similarly, β has c_ℓ poles of multiplicity ℓ and the denominator consists of factors q_ℓ^ℓ, where q_ℓ is a polynomial of degree c_ℓ. Finally, $\beta - 1$ has b_ℓ roots of multiplicity ℓ, hence its numerator has factors r_ℓ^ℓ, with $\deg r_\ell = b_\ell$. This determines β up to a constant k and leads to the following identities:

$$\beta(x) = \frac{\prod_\ell p_\ell^\ell}{k \prod_\ell q_\ell^\ell}, \qquad \beta(x) - 1 = \frac{\prod_\ell r_\ell^\ell}{k \prod_\ell q_\ell^\ell}.$$

Combining these identities and setting $p(x) = \prod_\ell p_\ell^\ell$, $q(x) = \prod_\ell p_\ell^\ell$, $r(x) = \prod_\ell r_\ell^\ell$,

$$p(x) - kq(x) = r(x). \tag{2.1}$$

As β is single-valued, and because by construction, each root of p, q, and r is distinct, there are no common factors among any pair of polynomials from $\{p_\ell\}_\ell \cup \{q_\ell\}_\ell \cup \{r_\ell\}_\ell$. Thus, all of the polynomials involved are pairwise relatively prime.

To proceed, Atkin & Swinnerton-Dyer's "differentiation trick" [2, 57] will be applied. That is, (2.1) will be differentiated and, in the case β does not represent a tree, k will be used to piece the equations together. Specifically, let p_0, q_0, r_0 be the greatest common divisors of $(p, p'), (q, q'), (r, r')$, respectively. Set

$$\begin{aligned} P &= \frac{p}{p_0}, \quad & Q &= \frac{q}{q_0}, \quad & R &= \frac{r}{r_0}, \\ \widetilde{P} &= \frac{p'}{p_0}, \quad & \widetilde{Q} &= \frac{q'}{q_0}, \quad & \widetilde{R} &= \frac{r'}{r_0}. \end{aligned}$$

Solving (2.1) for k and substituting into $p' - kq' = r'$,

$$\begin{aligned} r' = p' - q'\frac{p - r}{q} &\iff qr' = p'q - pq' + q'r \\ &\iff qr' - q'r = p'q - pq' \\ &\iff q_0 r_0 (Q\widetilde{R} - \widetilde{Q}R) = p_0 q_0 (\widetilde{P}Q - P\widetilde{Q}) \\ &\iff r_0 (Q\widetilde{R} - \widetilde{Q}R) = p_0 (\widetilde{P}Q - P\widetilde{Q}). \end{aligned}$$

As mentioned previously, p, hence p_0, is relatively prime to r, hence to r_0, so that by unique factorization, $p_0 \mid Q\widetilde{R} - \widetilde{Q}R$ and $r_0 | \widetilde{P}Q - P\widetilde{Q}$.

It remains to show that, up to a scalar,

$$p_0 = Q\widetilde{R} - \widetilde{Q}R, \qquad r_0 = \widetilde{P}Q - P\widetilde{Q}. \tag{2.2}$$

This will be shown by degree considerations. Because $\deg p_0 = \sum_{x \in \beta^{-1}(0)} \operatorname{mult}_x(p) - 1$, and similarly for q_0, r_0, and because $n := \deg \beta = \deg q = \deg r$, by the Riemann-Hurwitz formula applied to β,

$$2 = 2n - \deg p_0 - \deg q_0 - \deg r_0 = \deg q + \deg r - \deg p_0 - \deg q_0 - \deg r_0.$$

As $\deg Q\widetilde{R} = -1 + \deg q - \deg q_0 + \deg r - \deg r_0 = \deg \widetilde{Q}R$, and as $Q\widetilde{R}$ and $\widetilde{Q}R$ are both monic,

$$\deg Q\widetilde{R} - \widetilde{Q}R = -2 + \deg q + \deg r - \deg q_0 - \deg r_0 = -2 + 2 + \deg p_0 = \deg p_0.$$

Finally equating coefficients in (2.2) results in a system of multivariate polynomials. Due to the freedom to compose β with a Möbius transformation, there are three degrees of freedom that may be considered before solving the system of polynomials. First, if x is the pole with greatest multiplicity, then moving x to ∞ eliminates one root in the denominator having largest multiplicity. Additionally, if desired, post-composing β with $1/x$ or $1/(1-x)$ allows one to ensure that it is indeed a pole which has greatest multiplicity over all points in the domain of β.

The choice to place a pole at ∞ leaves two degrees of freedom, which allow one to specialize two coefficients of the polynomial system. For example, if p_ℓ has roots $\alpha_1, \ldots, \alpha_{\ell-1}$, then for any a, choosing $a\alpha_\ell = a(\alpha_1 \cdots \alpha_{\ell-1})^{-1}$ forces the constant term of p_ℓ to be a. In order to simplify the system as much as possible, the following specialization seems to work well in practice. After moving the pole of greatest multiplicity to ∞, determine a factor f from $\{p_\ell\}_\ell \cup \{q_\ell\}_\ell \cup \{r_\ell\}_\ell$ such that f has the greatest multiplicity, denoted m. If $\deg f = 1$, so that there is a unique point x with multiplicity m and image $\beta(x)$, then

1. set the constant term of f to be 0,

2. among the remaining factors of $\{p_\ell\}_\ell \cup \{q_\ell\}_\ell \cup \{r_\ell\}_\ell$, choose one with largest multiplicity and set the constant term to be 1.

On the other hand, if $\deg f > 1$, then

1. set the constant term of f to be 0 and the coefficient of the linear term to be 1.

As an alternative, choosing the constant term of the factor with largest multiplicity to be 0 and choosing a pole of largest multiplicity to lie at 1 can sometimes reduce the number of parasitic solutions (see [57, pg. 85-86]) of the system.

Additionally, because when computing this system, p_0, r_0 and $Q\widetilde{R} - \widetilde{Q}R, \widetilde{P}Q - P\widetilde{Q}$ may not end up with the same leading coefficients, multiplying p_0, r_0 by the leading coefficient of the corresponding polynomial may be necessary. Finally, having solved the polynomial system, k is given by $\big(p(x) - r(x)\big)/q(x)$ and can be recovered by evaluating this polynomial at any x which is neither a root nor a pole (see (2.1)).

2.1.2. Recognizing the Roots

Having constructed a multivariate polynomial system, it will be solved numerically and the approximate roots recognized using the LLL lattice basis reduction algorithm [40], in particular using the method in [31]. Either numerical Gröbner bases [36] or homotopy continuation methods [4] can efficiently solve such systems to arbitrary precision. The LLL lattice basis reduction algorithm is used to find a basis of short vectors of a lattice, with the first vector of the basis guaranteed to be particularly short.

Proposition 2.10. *[40, Prop. 1.11] Let $L \subseteq \mathbb{R}^n$ be a lattice with reduced basis $b_1, b_2, \ldots, b_n$. Then*

$$|b_1| \leq 2^{(n-1)/2}|x|$$

for every nonzero vector $x \in L$.

Algorithm 2.9 Constructing a polynomial system from a passport

```
function PassportToSystem($t_0$, $t_1$, $t_\infty$)
decrease the multiplicity of the largest part of $t_\infty$ by 1    ▷ move pole to $\infty$
polys ← []
for ($t$,s) in [($t_0$,'a'), ($t_1$,'b'), ($t_\infty$,'c')] do
    poly ← 1
    for $\ell^{a_\ell}$ in $t$ do
        factor ← $x^{a_\ell} + s_{\ell,a_\ell-1}x^{a_\ell-1} + \cdots + s_{\ell,0}$
        poly *= factor$^\ell$
    append poly to polys
$p, q, r$ ← polys
$p_0, q_0, r_0$ ← map(gcd, [(p,p’), (q,q’), (r,r’)])
coeffs1 ← coefficients($c\,r_0 - \widetilde{P}Q + P\widetilde{Q}$), $c$ = leading coefficient of $Q\widetilde{P} + P\widetilde{Q}$
coeffs2 ← coefficients($c\,p_0 - Q\widetilde{R} + \widetilde{Q}R$), $c$ = leading coefficient of $Q\widetilde{R} + R\widetilde{Q}$
▷ two smallest degree terms of largest multiplicity factors get coefficients $0, 1$:
Specialize(polys)
return concatenation of coeffs1, coeffs2
```

In order to recover an algebraic number α of degree at most d from a numerical approximation a, for $1 \leq n \leq d$, form the matrix

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 2^s & 0 \\ 0 & 1 & 0 & \dots & 0 & 2^s \operatorname{Re} a & 2^s \operatorname{Im} a \\ 0 & 0 & 1 & \dots & 0 & 2^s \operatorname{Re} a^2 & 2^s \operatorname{Im} a^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 2^s \operatorname{Re} a^n & 2^s \operatorname{Im} a^n \end{bmatrix} \tag{2.3}$$

and consider the lattice L generated by the rows $\{b_i\}_i$ of this matrix. The norm on this lattice is given by

$$\begin{aligned}\Big|\sum_i g_i b_i\Big| &= g_0^2 + g_1^2 + \cdots + g_n^2 + 2^s \operatorname{Re}\sum_i g_i a^i + 2^s \operatorname{Im}\sum_i g_i a^i \\ &= |g|^2 + 2^{2s}\Big|\sum_i g_i a^i\Big|\end{aligned}$$

On input the rows $\{b_i\}_i$ of the matrix, the LLL algorithm [40] will output a short vector $\sum_i g_i b_i$. As $\sum_i g_i b_i$ is short, $(1/2^{2s})\sum_i g_i b_i$ will be very short, forcing

$$\begin{aligned}&\operatorname{Re}(g_0 + g_1 a + \cdots + g_d a^d) \approx 0,\\ &\operatorname{Im}(g_0 + g_1 a + \cdots + g_d a^d) \approx 0.\end{aligned}$$

Definition. *The height of a polynomial $\sum_i a_j x^j$ is $\max_j |a_j|$. The height of an algebraic number is the height of its minimal polynomial.*

It is shown in [31] that for a polynomial g of degree at most d with $g(\alpha) \neq 0$, there is a lower bound on $|g(\alpha)|$. Hence, if $g(a)$ is smaller than this bound, for a sufficiently accurate approximation a to α, then necessarily $g(\alpha) = 0$. In particular,

Theorem 2.11. *[31, Thm. 1.15, Alg. 1.16] Let α be a complex number with $|\alpha| \leq 1$ and with minimal polynomial h of degree at most $d \geq 1$ and height at most H. Suppose that $|a| \leq 1$ and $|\alpha - a| \leq 2^{-s}/(4d)$, where s is the smallest positive integer such that*

$$2^s \geq 2^{d^2/2} \cdot (d+1)^{(3d+4)/2} \cdot H^{2d}. \tag{2.4}$$

Suppose that the basis reduction algorithm on input the rows of (2.3) yields a reduced basis with $\tilde{v} := \sum_i v_i b_i$ as the first vector. Then

$$|\tilde{v}| \leq 2^{d/2}(d+1)H$$

iff $v(\alpha) = 0$, where $v(x) = \sum_i v_i x^i$.

To find the minimal polynomial of α, it is then a matter of applying the theorem for $1 \leq n \leq d$ until the minimal polynomial is recovered. The reduction from $|\alpha| \geq 1$ to $|\alpha| \leq 1$ is handled by Explanation 1.16 in [31].

Assume that the result of LLL-reducing (2.3) is B, but no minimal polynomial of degree n is found. By inserting a column of zeros after the $(n+1)$th column and placing

$$\begin{bmatrix} 0 & \cdots & 0 & 1 & 2^s \operatorname{Re} a^{n+1} & 2^s \operatorname{Im} a^{n+1} \end{bmatrix}$$

after the last row, the first $n+1$ rows are already LLL-reduced, thus requiring less work to LLL-reduce the enlarged B than using (2.3) with n incremented.

Algorithm 2.12 Recognizing an algebraic number from an approximation

```
function AlgebraicRelation($a$, degree, height)
determine the largest integer $s$ satisfying (2.4)
if precision of $a$ is less than $2^{-s}/(4d)$ then
    Raise insufficient precision error
form the matrix $B$ as in (2.3) with $n = 1$
for $1 \le i \le d$ do
    apply LLL to $B$
    form $v$ as in Theorem 2.11
    if $|v| + 2^{2s}|\alpha|^2 < 2^{\texttt{degree}/2} \cdot (\texttt{degree} + 1) \cdot \texttt{height}$ then
        return $v$
    DimensionIncrease(B, [$\operatorname{Re} 2^s a^{i+1}$, $\operatorname{Im} 2^s a^{i+1}$])
Raise no algebraic relations with degree < degree and height < height
```

Having recognized all of the algebraic numbers in the solutions to the polynomial system, the resulting Belyĭ maps should be checked for the presence of parasitic solutions and for repeated isomorphism classes of constellations. The latter will occur whenever the degree of the field of definition of a Belyĭ map is larger than its Galois orbit. The end result is Algorithm 2.13.

2.1.3. Degree and Height Bounds

The one caveat of Theorem 2.11 is the requirement to have a bound on the degree and height of the minimal polynomial. In order to get a bound on the degree, one can use class multiplication coefficients (see Section 1.4.1.) [14, pg. 111-112].

Algorithm 2.13 Finding the classes of Belyĭ maps with a given passport

```
function PassportToBelyĭ($t_0$, $t_1$, $t_\infty$, degree, height)
belyis ← []
sys ← PassportToSystem($t_0$, $t_1$, $t_\infty$)
determine the largest integer $s$ satisfying (2.4)
sols ← solve(sys, prec=$2^{-s}/(4d)$)
for sol in sols do
    coeffs ← []
    for coeff in sol do
        min_poly ← AlgebraicRelation(coeff, degree, height)
        alg_coeff ← embedding in $\mathbb{Q}^{\mathrm{a}} \subseteq \mathbb{C}$ of approximate root coeff with minimal polynomial min_poly
        append alg_coeff to coeffs
    append to belyis the Belyĭ map of shape $(t_0, t_1, t_\infty)$ with coefficients coeffs
remove from belyis any Belyĭ maps whose numerator or denominator has incorrect degree, as well as all representatives except one for each Galois orbit
return belyis
```

However, this approach often leads to a larger than necessary bound on the degree, particularly when the combinatorial orbit splits into multiple Galois orbits. A less rigorous approach is to use the size of the combinatorial orbit, as obtained from `PassportToConstellations`, as a guide, noting that the minimal degree of a field of definition may be larger than the size of the combinatorial orbit [14, pg. 96]. However, using `PassportToConstellations` begins to become impractical for some passports around degree 13.

On the other hand, there is not currently a method to bound the height of the minimal polynomial of the field of definition, although it is hopeful that the work of Javanpeykar [29] can be used to obtain such a bound [33].

2.2. Belyĭ Maps to Passports

This is accomplished by first applying `Monodromy` (see Section 3.1.), followed by `ConstellationToPassport`.

Algorithm 2.14 Finding the passport of a Belyĭ map

```
function BelyĭToPassport($\beta$)
return ConstellationToPassport(Monodromy($\beta$))
```

3. Belyĭ Maps and Constellations

3.1. Belyĭ Maps to Constellations

The method discussed in this section was presented for genus zero by Schneps in [55] and for higher genera by Goins in [23]. A discussion of its convergence can be found in [55]. See also [17].

Let X be a compact Riemann surface and choose an embedding $X \hookrightarrow \mathbb{P}^r(\mathbb{C})$ defined by the polynomials $f_1, \ldots, f_{r-1} \in \mathbb{Q}^{\mathrm{a}}[x_1, \ldots, x_r]$. Let $\beta := p/q$ be a Belyĭ map of degree d on X and given $z_0 \in (0,1)$, let

$$Z := \beta^{-1}(z_0) = \{z_1, \ldots, z_d\}.$$

Fix a choice of z_0 so that $Z \subseteq \mathbb{A}^r_{\mathbb{C}}$. By definition, the monodromy representation of a Belyĭ map β of degree d is the group homomorphism

$$\rho : \pi_1(\mathbb{P}^1(\mathbb{C}) \backslash \{0, 1, \infty\}, z_0) \to S_d$$

determined by mapping a loop $g \in \pi_1(\mathbb{P}^1(\mathbb{C}) \backslash \{0, 1, \infty\}, z_0)$ to the permutation which sends the initial point of any lifting $\tilde{g}$ of g to the final point of $\tilde{g}$. If

$$g^{(a)} := a + (z_0 - a)e^{2\pi i t} \text{ for } a = 0, 1,$$

then $\pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}, z_0) = \langle g^{(0)}, g^{(1)}\rangle$ and ρ is determined by $\rho(g^{(0)})$ and $\rho(g^{(1)})$. As $\rho(g^{(a)})$ can be ascertained from the values $\tilde{g}_j(1)$, where $\tilde{g}_j$ is the lifting of $g^{(a)}$ beginning at z_j, the monodromy of β can be computed by finding the liftings $\tilde{g}_j$ for $a = 0$ and $a = 1$.

Let $g \subseteq \mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$ be a loop and let $\tilde{g} \subseteq X$ be a lifting of g by β. Writing $\tilde{g}(t) = \Big(g_1(t), \ldots, g_r(t)\Big)$, differentiation of $g = \beta\tilde{g} = \beta\Big(g_1(t), \ldots, g_r(t)\Big)$ implies that

$$\frac{dg}{dt} = \frac{d\beta\tilde{g}}{dt} = \sum_{j=1}^{r} \frac{\partial\beta}{\partial x_j} \cdot \frac{dg_j}{dt}. \tag{2.5}$$

Differentiating $f_k\tilde{g} = 0$, $1 \leq k \leq s$ yields

$$\sum_{j=1}^{s} \frac{\partial f_k}{\partial x_j} \cdot \frac{dg_j}{dt} = 0. \tag{2.6}$$

Returning to the case $z_0 \in (0,1)$ and $g = g^{(0)}$ or $g = g^{(1)}$,

$$\frac{dg^{(a)}}{dt} = 2\pi i(z_0 - a)e^{2\pi i t} = 2\pi i(g^{(a)} - a) = 2\pi i(\beta\tilde{g} - a).$$

Then (2.5) and (2.6) results in the matrix equation

$$\begin{bmatrix} \frac{\partial\beta}{\partial x_1}(\tilde{g}) & \cdots & \frac{\partial\beta}{\partial x_r}(\tilde{g}) \\ \frac{\partial f_1}{\partial x_1}(\tilde{g}) & \cdots & \frac{\partial f_1}{\partial x_r}(\tilde{g}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{r-1}}{\partial x_1}(\tilde{g}) & \cdots & \frac{\partial f_{r-1}}{\partial x_r}(\tilde{g}) \end{bmatrix} \begin{bmatrix} \frac{dg_1}{dt} \\ \vdots \\ \frac{dg_r}{dt} \end{bmatrix} = \begin{bmatrix} 2\pi i(\beta(\tilde{g}) - a) \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{2.7}$$

Setting $z_0 = 1/2$ is usually sufficient, and then supplying the condition $\tilde{g}(0) = z_j$ for $z_j \in Z$ results in an initial value problem which can be solved numerically. In particular, the resulting initial value problem is a first order autonomous system and can be efficiently solved.

When $X = \mathbb{P}^1(\mathbb{C})$, the initial value problem can be simplified. In this case, β is independent of x_1 and $f_0(x_0, x_1) = x_1$, which implies that $g_1 = 0$. Thus, (2.7) becomes

$$\begin{bmatrix} 2\pi i\Big(\beta(g_0, 0) - a\Big) \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{\partial\beta}{\partial x_0}(g_0, 0) & 0 \\ 0 & (g_0, 0) \end{bmatrix} \begin{bmatrix} \frac{dg_0}{dt} \\ 0 \end{bmatrix},$$

or $g_0' = 2\pi i(\beta g_0 - a)/(\beta' g_0)$.

Having found a lifting $\tilde{g}$ of $g^{(a)}$ with $\tilde{g}(0) = z_j$, the image of j under $\rho(g^{(a)})$ is then k such that

$$|z_k - \tilde{g}(1)| = \min_{1\leq \ell \leq d}(|z_\ell - \tilde{g}(1)|).$$

This leads to the following algorithm.

Algorithm 2.15 Computing monodromy numerically from (β, X)

```
function Monodromy($\beta$, $[f_1, \ldots, f_{r-1}]$)
$\sigma_0 \leftarrow$[0,...,0], $\sigma_1 \leftarrow$[0,...,0], $d \leftarrow \deg\beta$
$J \leftarrow \begin{bmatrix}\operatorname{grad}\beta & \operatorname{grad} f_1 & \ldots & \operatorname{grad} f_{r-1}\end{bmatrix}^T$
preimages$\leftarrow$solve($[\beta - 0.5, f_1, \ldots, f_{r-1}], [x_0, \ldots, x_{r-1}]$)
rhs(a)$\leftarrow \begin{bmatrix}2\pi i(\beta - a) & 0 & \ldots & 0\end{bmatrix}^T$
for $0 \leq j \leq d-1$ do
    $z_j \leftarrow$preimages[j]
    for $0 \leq a \leq 1$ do
        $\tilde{g} \leftarrow$solve($g' = \left(J^{-1}\cdot\texttt{rhs}(a)\right)(g)$, $g(0) = z_j$, $0 \leq t \leq 1$)
        index$\leftarrow$ $k$ such that |preimages[$k$]$-\tilde{g}(1)$| is minimized
        $\sigma_a$[$j$]$\leftarrow$index
return $\sigma_0$, $\sigma_1$
```

3.2. Constellations to Belyĭ Maps

A direct method which works for arbitrary genera is found in [33]. Presented here is a method which makes use of `ConstellationToPassport` and `PassportToBelyis`, hence this method is restricted to Belyĭ maps of genus zero. The only remaining consideration is to determine which Belyĭ map in the combinatorial orbit defined by the passport corresponds to the desired constellation. Thus, `Monodromy` and `CanonicalConstellation` are also utilized.

Algorithm 2.16 Finding a Belyĭ map from a constellation

```
function ConstellationToBelyĭ($\sigma_0$, $\sigma_1$, $\sigma_\infty$)
canonical ← CanonicalConstellation($\sigma_0$,$\sigma_1$,$\sigma_\infty$)
passport ← ConstellationToPassport($\sigma_0$,$\sigma_1$,$\sigma_\infty$)
belyis ← PassportToBelyis(passport)
for $\beta$ in belyis do
    constellation ← CanonicalConstellation( Monodromy($\beta$) )
    if constellation == canonical then
        return $\beta$
```

4. Constellations and Dessins

4.1. Dessins to Constellations

In order to recover a constellation corresponding to a dessin G, one begins by labeling the edges of G and forming a bipartition of the vertices. For each subset of the bipartition and each vertex in the subset, a cycle is formed by reading the edges around v in a sequential order. Collecting the cycles for all vertices of each subset of the bipartition gives the pair of permutations σ_0, σ_1.

4.2. Constellations to Dessin

Going from a constellation to a dessin is simply a matter of determining the white and black vertices which are connected by each edge and then using the constellation to order the edges around each vertex. Having identified the white vertices to which each edge is incident, given a cycle of σ_0, one proceeds through the cycle, for each edge determining the white vertex to which it is incident. Listing the white vertices in this order gives the cyclic ordering around the black vertex corresponding to that cycle. Repeating this for each cycle of σ_0 and similarly for σ_1 constructs the dessin.

This algorithm is in fact much simpler than its pseudocode suggests. Written in Python, Lines 3-20 become just 6 lines.

Algorithm 2.17 Finding a constellation from a dessin

```
function DrawingToConstellation($G$)
let (black, white) be a bipartition of $G$
label the edges of $G$ using $\{1, \ldots, n\}$
for color,branch in [(black,0), (white,1)] do
    permutation ← []
    for each vertex $v$ in color do
        cycle ← []
        for each sequential neighbor $w$ of $v$ do
            append the label of edge $(v, w)$ to cycle
        append cycle to permutation
    permutation ← the permutation defined by the cycles in permutation
    if branch == 0 then $\sigma_0$ ← permutation else $\sigma_1$ ← permutation
return $\sigma_0, \sigma_1$
```

Algorithm 2.18 Finding the dessin of a constellation

```
function ConstellationToDrawing($\sigma_0, \sigma_1$)
deg ← Max(MaxMoved($\sigma_0$, $\sigma_1$))
label the cycles of $\sigma_0$ from 1 to $m$
label the cycles of $\sigma_1$ from $m + 1$ to $n$
incident_to_which_0,incident_to_which_1 ← [],[]
for each (cycle, label) of $\sigma_0$ do
    for each edge in cycle append (edge,label) to incident_to_which_0
for each (cycle, label) of $\sigma_1$ do
    for each edge in cycle append (edge,label) to incident_to_which_1
black,white ← [],[]
for each (cycle, label) of $\sigma_0$ do
    rotation ← []
    for each edge in cycle do
        find edge in incident_to_which_1 and append label to rotation
    append (label, rotation) to black
for each (cycle, label) of $\sigma_1$ do
    rotation ← []
    for each edge in cycle do
        find edge in incident_to_which_0 and append label to rotation
    append (label, rotation) to white
return (black, white)
```

CHAPTER 3. COUNTING

1. Enumeration

For small degrees, the number of equivalent constellations can be counted simply by enumerating them. Given a degree and a genus, it is then simply a matter of generating the equivalence classes of passports for that degree and genus, applying `PassportToConstellations` to each equivalence class, and counting the number of constellations in each combinatorial orbit. In order to avoid overcounting equivalence classes of constellations, it is important to take care in how the passports are generated. First, because the entries in a representative of an equivalence class of passports can be reordered without changing the equivalence class, the passports of degree n will be generated by taking combinations of $\mathcal{P}_n$, the partitions of n. But, because a passport is a multiset, these combinations must be taken with replacement, what will be referred to as a multicombination.

As `PassportToConstellations` uses only two of the entries of the passport when computing the double cosets, and computation of $C_{S_n}(g_1)\backslash S_n/C_{S_n}(g_2)$ is easier for smaller conjugacy classes, it is desirable to have the largest conjugacy class in the last position of the passport. Based on the algorithm which Python uses to generate combinations, this can be achieved for most passports by reversing the usual order for generating partitions, instead generating the partitions in the order $(1, \ldots, 1), \ldots, (n)$. Although `PassportToConstellations` will reorder the passport to ensure the passport is ordered by increasing size of conjugacy classes, reversing the partitions will reduce the burden of such reordering. The result is Algorithms 3.1 and 3.2.

The number of equivalence classes of constellations of a given degree and genus, for degrees up to 12, were computed and appear in Table 3.1.

Algorithm 3.1 Counting the constellations of degree and genus

```
function CountConstellations(deg, gen)
count ← 0
for passport in PassportsOfDegreeAndGenus(deg, gen) do
    for constellation in PassportToConstellations(passport) do
        count += 1
return count
```

Algorithm 3.2 Generating the passports of degree and genus

```
function PassportsOfDegreeAndGenus(deg, gen)
passports ← []
partitions ← Reversed(Partitions(deg))
triples ← CombinationsWithReplacement(partitions, 3)
for triple in triples do
    if sum(map(length, triple)) − deg = 2 − 2gen then
        append triple to passports
return passports
```

A downside of the above algorithm is that many equivalence classes are generated multiple times, even though they are only counted once. For example,

```
PassportToConstellations([2, 2, 1], [3, 2], [3, 2])
```

finds the constellations

$$[(2,3)(4,5),(1,2,4)(3,5),(1,4,3)(2,5)], \quad [(2,3)(4,5),(1,4)(2,3,5),(1,4,3,5)].$$

However, since the second constellation has passport $(2^2 1, 3.2, 4.1) \neq (2^2 1, 3.2, 4.1)$, it is discarded, only to be generated again while performing

```
PassportToConstellations([2, 2, 1], [3, 2], [4, 1]).
```

Table 3.1.: Number of constellations by degree and genus for degree ≤ 12

Degree	Genus 0	Genus 1	Genus 2	Genus 3	Genus 4	Genus 5
1	1					
2	1					
3	2	1				
4	6	2				
5	14	9	4			
6	63	70	16			
7	269	443	182	30		
8	1336	3255	2245	385		
9	6988	23971	23895	7450	900	
10	39304	177247	256041	131928	19344	
11	224640	1326642	2660722	1996108	516100	54990
12	1334227	10041819	26706878	28872190	12177550	1588218

But $[(2,3)(4,5),(1,4)(2,3,5),(1,4,3,5)]$ is the only constellation with passport $(2^2 1, 3.2, 4.1)$, so that there is no reason to run

```
PassportToConstellations([2, 2, 1], [3, 2], [4, 1]),
```

since this constellation can be found during the computation

```
PassportToConstellations([2, 2, 1], [3, 2], [3, 2]).
```

Because only two entries of the passport are used in the computation of the double coset, it makes sense to loop over pairs of partitions instead of triples of partitions. This way, the double cosets for

$$C_{S_5}\Big((1\,2)(3\,4)\Big)\backslash S_5/C_{S_5}\Big((1\,2\,3)(4\,5)\Big)$$

are computed only once, rather than being computed for each passport of the form $(2^2 1, 3.2, \cdot\,)$. When using this approach, all genera are counted simultaneously in order to preserve the ability to compute the double cosets only once.

Although the number of double cosets being computed has been reduced, constellations are still generated multiple times. The constellation with passport $(2^2 1, 3.2, 4.1)$ is generated from the double cosets of both

$$C_{S_5}\Big((1\,2)(3\,4)\Big)\backslash S_5/C_{S_5}\Big((1\,2\,3)(4\,5)\Big) \quad \text{and} \quad C_{S_5}\Big((1\,2)(3\,4)\Big)\backslash S_5/C_{S_5}\Big((1\,2\,3\,4)\Big).$$

In order to ensure that constellations are not included multiple times, a constellation will only be counted when its passport is in nondecreasing order[1]. For the above example, $[(2\,3)(4\,5), (1\,4)(2\,3\,5)]$ would be counted only when it is generated from the pair of cycle types $(2^2 1, 3.2)$, in which case its passport would be $(2^2 1, 3.2, 4.1)$. But the set of 2-multicombinations from $\mathcal{P}_n$ still turns out to be redundant and many of the 2-multicombinations can be excluded.

Lemma 3.3. *The only constellations with passport of the form* $(1^n, \cdot, \cdot)$ *is the constellation with passport* $(1^n, n, n)$.

Proof. If $t_0 = 1^n$ and x has cycle type t_0, then $x = 1$ and $\langle x, y\rangle$ transitive forces y to be an n-cycle. Moreover, $(xy)^{-1} = y^{-1}$ is also an n-cycle. □

Proposition 3.4. *Let* $\mathcal{P}_n$ *denote the set of partitions of* n *and let* $\mathbb{O}$ *be the constellations output by performing* `PartialPassportToConstellations` *on all* 2*-multicombinations from* $\mathcal{P}'_n := \mathcal{P}_n \backslash \{(1^n, n)\}$*, as well as on* (n, n).

1. *All equivalence classes of constellations of degree* n *are in* $\mathbb{O}$.
2. *Each equivalence class of constellations generated by a* 2*-multicombination from* $\mathcal{P}'_n$ *appears in* $\mathbb{O}$ *with nondecreasing passport only once.*
3. *Each equivalence class of constellations generated by* (n, n) *appears in* $\mathbb{O}$ *only once.*

[1] The entries are compared lexicographically, where the absence of an element is less than any element. Thus, $[6, 4, 2] < [7, 3]$ and $[3, 2] < [3, 2, 1]$. Further, it is assumed that each entry of a passport is sorted in decreasing order, as is standard for a partition.

Proof. (1),(3) Let Δ be a constellation of degree n. Given the passport (t_0, t_1, t_i) of Δ, rearrange (t_0, t_1, t_i) so that $t_0 \leq t_1 \leq t_i$. If $t_1 \neq n$, then $t_0 \neq n$ by ordering and $t_0 \neq 1^n$ by Lemma 3.3. That is, (t_0, t_1) is a 2-multicombination from $\mathcal{P}'_n$ and Δ is generated by `PartialPassportToConstellations` on input (t_0, t_1).

If $t_1 = n$, then $t_i = n$ and Δ is generated by `PartialPassportToConstellations` on input (n, n) and was not generated by any 2-multicombination from $\mathcal{P}_n$.

(2) The 2-multicombination (t_0, t_1) generates Δ with passport (t_0, t_1, t_i), which is in nondecreasing order. If Δ is generated by `PartialPassportToConstellations` on input $(t'_0, t'_1) \neq (t_0, t_1)$, then either $t'_0 \neq \min(t_0, t_1, t_i)$ or $t'_1 \neq \min(t_1, t_i)$. In either case, the passport of the generated constellation is not nondecreasing and Δ is generated with nondecreasing passport only once.

□

The above observations are then implemented as

Algorithm 3.5 Counting equivalence classes of constellations of a given degree

```
function CountMonodromies(deg)
counts ← 0
partitions ← Reversed( Partitions(deg)$\setminus\{[1,\dots,1],[n]\}$ )
pairs ← CombinationsWithReplacement(partitions, 2)
for pair in pairs do
    constells ← PassportToConstellations(pair)
    for $(s_0, s_1, s_\infty)$ in constells do
        if HasSortedPassport($s_0, s_1, s_\infty$) then
            counts[Genus($s_0, s_1, s_\infty$)] += 1
return counts
```

2. Counting Trees

Let $\Delta = (\sigma_0, \sigma_1, \sigma_\infty)$ be a constellation with passport (t_0, t_1, t_∞). Let S_n act on itself by conjugation, $x^h := h^{-1}xh$, and on $S_n \times S_n$ by simultaneous conjugation,

$(x,y)^h := (x^h, y^h)$. By simultaneously conjugating Δ, it be can be assumed that $\sigma_\infty = g^{-1}$, where g is an arbitrary but fixed element of $\mathcal{C}_{t_\infty}$. Then every constellation with passport (t_0, t_1, t_∞) is isomorphic to an element of the set

$$\Sigma := \{(x,y) \in \mathcal{C}_{t_0} \times \mathcal{C}_{t_1} \mid xy = g\}.$$

That is, the class multiplication coefficient $c_{t_0,t_1}^{t_\infty} = \#\Sigma$ (Section 1.4.1.) gives an upper bound on the number of constellations with passport (t_0, t_1, t_∞). However, many of the elements of Σ may be isomorphic under simultaneous conjugation. The question then becomes, how many elements of Σ are isomorphic to a given pair (x,y)?

Lemma 3.6. *Let $C_{S_n}(g)$ act on Σ by simultaneous conjugation. Then the constellations with passport (t_0, t_1, t_∞) are in bijection with the orbits of Σ under this action.*

Proof. In order that $(x,y)^\tau \in \Sigma$, it must be that $\tau \in C_{S_n}(g)$. The representatives of the constellation (x,y) in Σ then correspond to the distinct images of (x,y) under simultaneous conjugation by $C_{S_n}(g)$, that is, the orbit of (x,y). □

It will now be shown that

Proposition 3.7. *For an arbitrary but fixed n-cycle $g \in S_n$, there are*

$$\frac{1}{n}\sum_{d|n} \varphi\left(\frac{n}{d}\right) |\mathrm{fix}(g^d)|$$

constellations with passport (t_0, t_1, n), where

$$\mathrm{fix}(h) := \{x \in \mathcal{C}_{t_0} \mid x^{-1}g \in \mathcal{C}_{t_1}, x^h = x\}.$$

Before proceeding, two observations are necessary. First, for $h \in C_{S_n}(g)$ and $(x,y) \in \Sigma$, if $x^h = x$,

$$y^h = (x^{-1}g)^h = (x^{-1})^h g^h = x^{-1}g = y,$$

so that h fixes (x,y) iff h fixes x. Additionally,

$$x^h = x \Longrightarrow (x^k)^{k^{-1}hk} = x^k, \qquad y^{k^{-1}hk} = y \Longrightarrow (y^{k^{-1}})^h = y^{k^{-1}}$$

so that there is a bijection $\mathrm{fix}(h) \longleftrightarrow \mathrm{fix}(h^k)$. In particular, $|\mathrm{fix}(\cdot)|$ is constant on conjugacy classes.

By the Cauchy-Frobenius lemma, the number of orbits $\mathcal{O}_i$ of $C_{S_n}(g)$ acting on Σ is given by

$$\#\{\mathcal{O}_i\}_i = \frac{1}{|C_{S_n}(g)|} \sum_{h \in C_{S_n}(g)} |\mathrm{fix}(h)| = \frac{1}{|C_{S_n}(g)|} \sum_{\vartheta \vdash n} |\mathcal{C}_\vartheta||\mathrm{fix}(h_\vartheta)|, \tag{3.1}$$

where $\{\mathcal{C}_\vartheta\}_{\vartheta \vdash n}$ are the conjugacy classes of $C_{S_n}(g)$ and h_ϑ is an element of cycle type ϑ.

In the case that Δ is a tree, so that σ_∞ is an n-cycle, more can be said.

Lemma 3.8. *If $k \in S_n$ is an n-cycle, then $C_{S_n}(k) = \langle k \rangle$.*

Proof. Observe that for $h \in C_{S_n}(k)$, where $k := (a_1 \cdots a_n)$,

$$h(a_i) = hk^{i-1}(a_1) = k^{i-1}h(a_1),$$

so that h is determined once $h(a_1)$ is determined. As there are at most n choices for $h(a_1)$, while $\langle k \rangle \leq C_{S_n}(k)$, it follows that $\langle k \rangle = C_{S_n}(k)$. □

Note that the cycle decomposition of a power s of an r-cycle h is given by d r/d-cycles, where $d = (r, s)$, and that there are $\varphi(r/d)$ elements of this cycle type in $\langle h \rangle$. Now let $g \in S_n$ be an n-cycle. For $h, k \in C_{S_n}(g)$ having the same cycle type, $\mathrm{fix}(h) = \mathrm{fix}(k)$. This is because if h, k have the same cycle type, then $h = g^a$ and $k = g^b$, where $(a, n) = (b, n)$. But then both h, k generate a subgroup of $C_{S_n}(g)$ of order $n/(a, n)$. As there is a unique subgroup of the cyclic group $C_{S_n}(g)$ of each order, $\langle h \rangle = \langle k \rangle$. Thus,

$$x^h = x \Longleftrightarrow h \in C_{S_n}(x) \Longleftrightarrow k \in C_{S_n}(x) \Longleftrightarrow x^k = x.$$

Combining these observations with (3.1) and Lemma 3.6 completes the proof of Proposition 3.7.

2.1. Counting Trees of Prime Degree

Proposition 3.9. *For p prime and $(t_0, t_1) \neq (1^p, \cdot), (p, p)$, there are $c^{(p)}_{t_0,t_1}/p$ constellations with passport equivalent to (t_0, t_1, p).*

Proof. Assume that Δ is a tree of prime degree p which does not have passport $(1^p, p, p)$ (see Lemma 3.3) or (p, p, p). One of t_0, t_1 is different from $1^p, p$, and it may be assumed that $t_0 \neq 1^p, p$. Let $x \in \mathcal{C}_{t_0}$ be such that $x^g = x$, implying $g^x = g$ and $x \in C_{S_p}(g)$. From Lemma 3.8, $C_{S_p}(g)$ consists of the identity and $p - 1$ p-cycles, but x is neither the identity nor a p-cycle. As this is a contradiction, it follows that $\operatorname{fix}(g) = \emptyset$ and the formula in Proposition 3.7 has only the single term corresponding to $d = p$. As 1 fixes every pair in Σ and $|\Sigma| = c^{(p)}_{t_0,t_1}$, □

2.2. Counting Constellations with Passport (n, n, n)

The formula given in this section was originally proven as Theorem 1 in [46] by specializing the formula given in Theorem 1 of [45]. Below is a direct and elementary proof which avoids the use of the character theory of the symmetric group.

Let $\Delta = (\sigma_0, \sigma_1, \sigma_\infty)$ be a constellation with passport (n, n, n) and let $g = (1\,2\,\cdots\,n)$. First, if n is even, then σ_0, σ_1 being n-cycles implies that they are odd permutations. As a result, σ_∞ must be even and cannot be an n-cycle. Thus, there are no constellations with passport (n, n, n) for n even.

Let n be an odd integer. By analyzing the structure of $C_{S_n}(g)$, it is possible to derive an explicit expression for $|\operatorname{fix}(g^d)|$. Before beginning, note that the number of ways of writing an n-cycle as a product of two n-cycles is [10, 12]

$$c^{(n)}_{(n),(n)} = \frac{2(n-1)!}{n+1}.$$

Now recall that g^d is a product of r n/r-cycles, where $r = (n, d)$. When a permutation is a product of cycles of a given length, its centralizer admits an explicit description as a wreath product.

Recall that the action of $g \in S_m$ on $b \in \operatorname{Fun}\Big(\{1, \ldots, m\}, \mathbb{Z}/\ell\mathbb{Z}\Big)$ is defined by $b^g(x) = b(x^{g^{-1}})$.

Lemma 3.10. *[60] Let σ be a product of m disjoint ℓ-cycles. Then*

$$C_{S_n}(\sigma) \approx (\mathbb{Z}/\ell\mathbb{Z}) \wr S_m = \mathrm{Fun}\Big(\{1, \ldots, m\}, \mathbb{Z}/\ell\mathbb{Z}\Big) \rtimes S_m.$$

Remark 2. If $\sigma = \prod h_i$, where h_i is the product of all cycles of length i in the disjoint cycle representation of σ, then $C_{S_n}(\sigma) = \prod C_{S_n}(h_i)$.

It is necessary to establish some notation and provide an explicit description of the isomorphism in the lemma. Let

$$\sigma := (a_{1,0} \ \cdots \ a_{1,\ell-1})(a_{2,0} \ \cdots \ a_{2,\ell-1}) \cdots (a_{m,0} \ \cdots \ a_{m,\ell-1}),$$

where the second index of $a_{i,j}$ will be considered modulo ℓ. Given $\eta \in C_{S_n}(\sigma)$, define $f_\eta \in \mathrm{Fun}(\{1, \ldots, m\}, \mathbb{Z}/\ell\mathbb{Z})$ and $\tau_\eta \in S_m$ by

$$\eta(a_{i,0}) = a_{j,k} \Longleftrightarrow \begin{cases} i^{\tau_\eta} = j, \\ f_\eta(j) = k. \end{cases} \qquad \Big(\text{Equivalently, } \eta(a_{i,0}) = a_{\tau(i), f(i\,\tau)}.\Big) \tag{3.2}$$

The isomorphism given in Lemma 3.10 is then $\eta \mapsto (f_\eta, \tau_\eta)$, and being a homomorphism,

$$\tau_{\eta\vartheta} = \tau_\eta \tau_\vartheta, \qquad f_{\eta\vartheta} = f_\eta + f_\vartheta^{\tau_\eta^{-1}}. \tag{3.3}$$

Lemma 3.11. *A permutation $\eta \in (\mathbb{Z}/\ell\mathbb{Z}) \wr S_m = C_{S_n}(\sigma)$ is an ℓm-cycle iff τ_η is an m-cycle and $(f_{\eta^m}(1), \ell) = 1$.*

Proof. Suppose the latter condition holds and that $\eta^t(a_{1,0}) = a_{1,0}$ for some $0 < t \leq \ell m$. Using (3.3),

$$1 = \tau_{\eta^t}(1) = \tau_\eta^t(1).$$

But τ_η is an m-cycle, so that $t = sm$ for some $m \in \mathbb{Z}_+$. Moreover, from (3.3) and the fact that $\tau_{\eta^m} = \tau_\eta^m = 1$,

$$f_{\eta^{sm}} \equiv f_{\eta^m} + f_{\eta^{sm-m}}^{\tau_{\eta^m}^{-1}} \equiv f_{\eta^m} + f_{\eta^{sm-m}} \quad (\ell). \tag{3.4}$$

By induction, $f_{\eta^{sm}} \equiv s f_{\eta^m} \;\; (\ell)$ and by (3.2),

$$0 \equiv f_{\eta^t}(1) \equiv f_{\eta^{sm}}(1) \equiv s f_\eta(1) \quad (\ell).$$

As $(f_\eta(1), \ell) = 1$, $s \equiv 0 \;\; (\ell)$ and $t = \ell m$. That is, η is an ℓm-cycle.

Suppose $\eta \in (\mathbb{Z}/\ell\mathbb{Z}) \wr S_m$ is an ℓm-cycle. Then

$$\{\eta^t(a_{1,0})\}_{t=1}^{\ell m} \longleftrightarrow \{a_{i,j}\}_{\substack{1 \leq i \leq m \\ 0 \leq j \leq \ell - 1}} \tag{3.5}$$

is a bijection. For all $1 \leq j \leq m$, $a_{j,0} = \eta^t(a_{1,0})$ for some $1 \leq t \leq \ell m$, and by (3.2) and (3.3),

$$j = \tau_{\eta^t}(1) = \tau_\eta^t(1)$$

for some $1 \leq t \leq \ell m$. Thus, the orbit of 1 under τ_η is $\{1, \ldots, m\}$ and τ_η is an m-cycle.

Because τ_η is an m-cycle, $\eta^t(a_{1,0}) = a_{1,j}$ iff $t \equiv 0 \;\; (m)$. From this and (3.5), $\{a_{1,j}\}_{j=0}^{\ell-1} \leftrightarrow \{\eta^{sm}(a_{1,0})\}_{s=0}^{\ell-1}$. But by (3.2) and (3.4), whose hypothesis was that τ be an m-cycle, $f_{\eta^{sm}} \equiv s f_{\eta^m} \;\; (\ell)$, so that

$$\{a_{1,j}\}_{j=0}^{\ell-1} = \{\eta^{sm}(a_{1,0})\}_{s=0}^{\ell-1} = \{a_{1, s f_\eta^m(1)}\}_{s=0}^{\ell-1}$$

implies that $f_{\eta^m}(1)$ generates $\mathbb{Z}/\ell\mathbb{Z}$ and $(f_{\eta^m}(1), \ell) = 1$. □

It will be necessary to have a formula for the image of $a_{i,0}$ under a power $(yx)^r \in C_{S_n}(\sigma)$.

Lemma 3.12. *Let* $y, x \in \operatorname{Fun}(Y, \mathbb{Z}/\ell\mathbb{Z}) \rtimes S_m$*. Then*

$$f_{(yx)^r} = \sum_{s=1}^{r} \left(f_y^{\tau_{yx}^{-s+1}} + f_x^{\tau_y^{-1} \tau_{yx}^{-s+1}} \right).$$

Proof. For the case $r = 1$, observe that $f_{yx} = f_y + f_x^{\tau_y^{-1}}$. Assume the hypothesis holds for $r = t - 1$ and note that

$$f_{(yx)^t} = f_y + f_{x(yx)^{t-1}}^{\tau_y^{-1}},$$

$$f_{x(yx)^{t-1}} = f_x + f_{(yx)^{t-1}}^{\tau_x^{-1}}.$$

Putting these together yields

$$\begin{aligned} f_{(yx)^t} &= f_y + f_{x(yx)^{t-1}}(\cdot\,\tau_y) \\ &= f_y + f_x(\cdot\,\tau_y) + f_{(yx)^{t-1}}^{\tau_x^{-1}}(\cdot\,\tau_y) \\ &= f_y + f_x^{\tau_y^{-1}} + \sum_{s=1}^{t-1} \Big(f_y^{\tau_{yx}^{-s+1}}(\cdot\,\tau_{yx}) + f_x^{\tau_y^{-1}\tau_{yx}^{-s+1}}(\cdot\,\tau_{yx}) \Big), \end{aligned}$$

and the result follows by induction. □

Definition. *[54] Let $\varphi_r(n)$ be the number of groups of r consecutive integers*

$$0 \le m < m+1 < \cdots < m+r-1 \le n-1$$

with $(m,n) = (m+1,n) = \cdots = (m+r-1,n) = 1$.

The functions $\varphi_r(n)$ counting the number of r consecutive integers relatively prime to n were considered by Victor Schemmel in [54], where it is noted that φ_r is multiplicative and given on prime powers by

$$\varphi_r(p^e) = p^{e-1}(p-r).$$

As $(0,1) = (1,1) = 1$, $\varphi_2(1) = 1$.

Theorem 3.13. *For n odd, there are*

$$\frac{2}{n} \sum_{d \mid n} \left(\frac{n}{d}\right)^{(d-1)} \varphi\left(\frac{n}{d}\right) \varphi_2\left(\frac{n}{d}\right) \frac{(d-1)!}{d+1} \tag{3.6}$$

constellations with passport (n,n,n).

Proof. It may be assumed that $n > 1$, since the case $n = 1$ was handled in Lemma 3.3. Let $d \mid n$ and $\ell := n/d$. From Proposition 3.7, it remains to determine $|\mathrm{fix}(g^d)|$, where $g = (1\ \cdots\ n)$, hence

$$g^d = (1\ \ d{+}1\ \cdots\ n{-}d{+}1)\cdots(d\ \ 2d\ \cdots\ n) =: (a_{1,0}\ \cdots\ a_{1,\ell-1})\cdots(a_{d,0}\ \cdots\ a_{d,\ell-1}).$$

By definition,

$$x \in \mathrm{fix}(g^d) \iff x^{(g^d)} = x,\ x \in \mathcal{C}_{(n)},\ x^{-1}g \in \mathcal{C}_{(n)}.$$

By Lemma 3.11, $x \in \mathcal{C}_{(n)} \cap C_{S_n}(g^d)$ iff τ_x is a d-cycle and $(f_{x^d}(1), \ell) = 1$. It will be shown that

$$x^{-1}g \in \mathcal{C}_{(n)} \cap C_{S_n}(g^d) \iff \tau_{x^{-1}g} \text{ is a } d\text{-cycle}, (f_{x^d}(1) - 1,\, \ell) = 1. \tag{3.7}$$

As a result,

$$\begin{aligned} x \in \operatorname{fix}(g^d) \iff x \in C_{S_n}(g^d),\ \tau_x \text{ is a } d\text{-cycle},\ (f_{x^d}(1), \ell) = 1,& \\ \tau_{x^{-1}g} \text{ is a } d\text{-cycle},\ (f_{x^d}(1) - 1, \ell) = 1.& \end{aligned}$$

The condition on $\tau_x, \tau_{x^{-1}g}$ is satisfied iff there are permutations $h, k \in \mathcal{C}_{(d)} \subseteq S_d$ so that $hk = \tau_g$. As τ_g is a d-cycle by Lemma 3.11, there are $c^{(d)}_{(d),(d)}$ such pairs h, k. The condition on $f_{x^d}(1) - 1, f_{x^d}(1)$ is equivalent to choosing $d-1$ arbitrary elements from $\mathbb{Z}/\ell\mathbb{Z}$ for $f_{x^d}(i)$, $1 < i \leq d$, along with a pair $m, m+1 \in (\mathbb{Z}/\ell\mathbb{Z})^\times$ for $f_{x^d}(i) - 1, f_{x^d}(i)$. As there are $\varphi_2(\ell)$ pairs $(m, m+1)$ and $\ell^{(d-1)}$ choices for the other values, this gives $\ell^{(d-1)}\varphi_2(\ell)$ functions satisfying the conditions on f_{x^d}. Then Applying the formulas for $c^{(d)}_{(d),(d)}$ and $\varphi_2(\ell)$, along with Proposition 3.7, will give the result.

It remains to prove (3.7). Let $x^{-1}g \in C_{S_n}(g^d) \cap \mathcal{C}_{(n)}$ and note that this is equivalent to $xg^{-1} \in C_{S_n}(g^d) \cap \mathcal{C}_{(n)}$. Before proceeding, $f_{g^{-1}}$ will be determined. As $a_{i,0} = i$, for $2 \leq i \leq d$, $g^{-1}(a_{i,0}) = g^{-1}(i) = i - 1$. On the other hand, $g^{-1}(a_{1,0}) = g^{-1}(1) = n = a_{d,\ell-1}$ and by (3.2)

$$g^{-1}(a_{i,0}) = \begin{cases} a_{d,\ell-1} & \text{if } i = 1, \\ a_{i-1,0} & \text{if } 2 \leq i \leq m, \end{cases} \implies f_{g^{-1}}(j) = \begin{cases} -1 & \text{if } j = d, \\ 0 & \text{if } 1 \leq j \leq d-1. \end{cases} \tag{3.8}$$

As $\tau_{g^{-1}}$, τ_x, and $\tau_{g^{-1}x}$ are d-cycles by Lemma 3.11,

$$\{1\,\tau_x^{s-1}\}_{s=1}^d = \{1, \ldots, d\} = \{1\,\tau_{g^{-1}x}^{s-1}\}_{s=1}^d = \{1\,\tau_{g^{-1}x}^{s-1}\tau_{g^{-1}}\}_{s=1}^d. \tag{3.9}$$

By Lemma 3.12 with $i = 1$, $y = g^{-1}$, and $r = d$, along with (3.8) and (3.9),

$$f_{(g^{-1}x)^d}(1) = \sum_{s=1}^d \Big(f_{g^{-1}}(1\,\tau_{g^{-1}x}^{s-1}) + f_x(1\,\tau_{g^{-1}x}^{s-1}\tau_{g^{-1}}) \Big) = \sum_{s=1}^d f_{g^{-1}}(s) + \sum_{s=1}^d f_x(s). \tag{3.10}$$

Taking $y = 1$ in Lemma 3.12 and using $f_1(j) = 0$ for all j shows that

$$f_{x^d}(1) = \sum_{s=1}^d \Big(f_1^{\tau_x^{-s+1}}(1) + f_x^{\tau_x^{-s+1}}(1) \Big) = \sum_{s=1}^d f_x(s). \tag{3.11}$$

Thus, by (3.10) and (3.11),

$$f_{(g^{-1}x)^d}(1) = \sum_{s=1}^{d} f_{g^{-1}}(s) + \sum_{s=1}^{d} f_x(s) \equiv -1 + f_{x^d}(1) \quad (\ell).$$

It follows that, given an n-cycle $x \in C_{S_n}(g^d)$,

$$\begin{aligned} x^{-1}g \in \mathcal{C}_{(n)} &\iff \tau_{g^{-1}x} \text{ is a } d\text{-cycle}, \left(f_{(g^{-1}x)^d}(1), \ell\right) = 1 \text{ by Lemma 3.11} \\ &\iff \tau_{g^{-1}x} \text{ is a } d\text{-cycle}, (f_{x^d}(1) - 1, \ell) = 1, \end{aligned}$$

as desired. □

2.3. Applications to Counting by Degree and Genus

First, note that by the Riemann-Hurwitz formula, for a constellation of degree n and genus g with Belyĭ map β,

$$\begin{aligned} 2g - 2 = -2n \; &+ \sum_{p \in \beta^{-1}(0)} \left(\mathrm{mult}_p(\beta) - 1\right) \; + \sum_{p \in \beta^{-1}(1)} \left(\mathrm{mult}_p(\beta) - 1\right) \\ &+ \sum_{p \in \beta^{-1}(\infty)} \left(\mathrm{mult}_p(\beta) - 1\right) \\ = n &- |\beta^{-1}(0)| - |\beta^{-1}(1)| - |\beta^{-1}(\infty)| \end{aligned}$$

since $\sum_{p \in \beta^{-1}(z)} \mathrm{mult}_p(\beta) = n$ for all $z \in \mathbb{P}^1(\mathbb{C})$. Rearranging,

$$n = 2g - 2 + |\beta^{-1}(0)| + |\beta^{-1}(1)| + |\beta^{-1}(\infty)|. \tag{3.12}$$

Lemma 3.14. *Every constellation of degree n and genus $\lceil (n-3)/2 \rceil$ is equivalent to a tree.*

Proof. From (3.12), if all of $|\beta^{-1}(0)|, |\beta^{-1}(1)|, |\beta^{-1}(\infty)| \geq 2$, then

$$n \geq 2\left\lceil \frac{n-3}{2} \right\rceil - 2 + 6 \iff \frac{n-4}{2} \geq \left\lceil \frac{n-3}{2} \right\rceil,$$

a contradiction. □

As a result of Proposition 3.9, this lemma says that for n an odd prime, the equivalence classes of constellations of degree n and genus $(n-3)/2$ can be counted simply by summing the class multiplication coefficients over the passports of genus $(n-3)/2$. This has been done in Table 3.2.

Table 3.2.: Number of constellations of Degree n, Genus $(n-3)/2$

n	13	17	19	23
#	54492480	1645049606400	430325616142080	574441148941712640000

Lemma 3.15. *There are no constellations of genus g with degree less than $2g+1$ and there are no constellations of degree n with genus greater than $(n-1)/2$. For n odd, all constellations of degree n and genus $(n-1)/2$ have passport (n,n,n).*

Proof. From (3.12), if $2g+1 \geq n$,

$$\begin{aligned} 2g+1 &\geq 2g-2+|\beta^{-1}(0)|+|\beta^{-1}(1)|+|\beta^{-1}(\infty)| \\ &\iff 3 \geq |\beta^{-1}(0)|+|\beta^{-1}(1)|+|\beta^{-1}(\infty)|. \end{aligned}$$

Thus, $2g+1 > n$ would imply that $|\beta^{-1}(z)| = 0$ for some $z \in \{0,1,\infty\}$, which contradicts the surjectivity of a holomorphic map from a compact Riemann surface. Moreover, if $n = 2g+1$, each of $0, 1, \infty$ can have only one point lying over them. □

Thus, the number of equivalence classes of constellations of odd degree n and genus $(n-1)/2$ are given by (3.6), the values of which are in Table 3.3.

Table 3.3.: Number of constellations of Degree n, Genus $(n-1)/2$

n	13	15	17	19	21
#	5263764	726485868	136750260720	33696703714374	10532043325452570

The previous two lemmas allow for the exclusion of either A_n or S_n as monodromy group when Δ has degree n, genus $\lfloor (n-1)/2 \rfloor$.

Proposition 3.16.

1. *There are no constellations of degree n, genus $(n-2)/2$ with* $\operatorname{Mon}\Delta = A_n$.

2. *There are no constellations of degree n, genus $(n-1)/2$ with* $\operatorname{Mon}\Delta = S_n$.

Proof. (i) By Lemma 3.14, any such constellation is equivalent to a tree, hence $\operatorname{Mon}\Delta$ contains an n-cycle for n even and $\operatorname{Mon}\Delta \neq A_n$.

(ii) By Lemma 3.15, any such constellation has $\operatorname{Mon}\Delta$ generated by n-cycles, where n is odd. As a result, $\operatorname{Mon}\Delta \leq A_n$. □

CHAPTER 4. COMPOSITION

Given a Belyĭ map γ and a dynamical Belyĭ map β, $\beta\gamma$ is again a Belyĭ map (see Section 1.3.2.1.). It turns out that the constellations Δ_β and Δ_γ are not enough to determine the constellation $\Delta_{\beta\gamma}$. In fact, $\Delta_{\beta\gamma}$ depends on the particular Belyĭ map β. However, if a small amount of additional information is provided about where the dessin of β lies in $\mathbb{P}^1(\mathbb{C})$, then $\Delta_{\beta\gamma}$ can be determined from this information along with the constellations Δ_β and Δ_γ.

Throughout this Chapter, β will denote a dynamical Belyĭ map and the situation under consideration will be $X \xrightarrow{\gamma} Y := \mathbb{P}^1(\mathbb{C}) \xrightarrow{\beta} Z := \mathbb{P}^1(\mathbb{C})$.

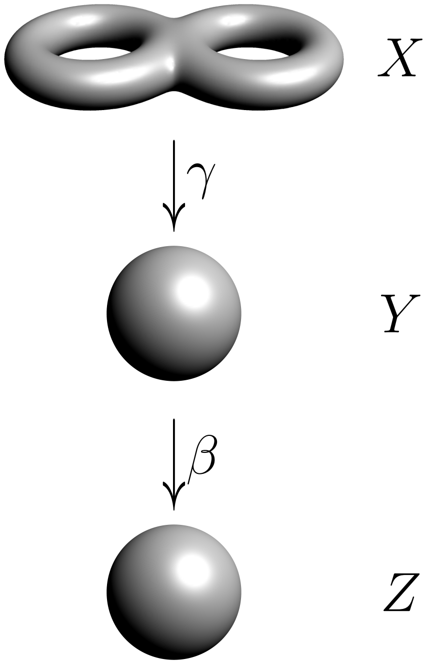


Figure 4.1.: A composition of covering maps

As a first observation in the determination of the constellation of $\beta\gamma$, γ is a covering map of Y away from 0, 1, and ∞, so that γ evenly covers $Y\backslash\{0, 1, \infty\}$. As

β is a dynamical Belyĭ map, $\{0,1,\infty\} \cap \beta^{-1}(0,1) = \emptyset$ and each of the $\deg \gamma$ sheets lying over Y contains a homeomorphic copy of $\beta^{-1}(0,1)$. Thus, $(\beta\gamma)^{-1}(0,1)$ consists of $\deg \gamma$ copies of $\beta^{-1}(0,1)$, the disjoint segments of the dessin of β. From this and the fact that edges of γ, E_γ are bijection with the sheets of γ, the edges of $\beta\gamma$, $E_{\beta\gamma}$, are in bijection with the Cartesian product $E_\gamma \times E_\beta$. Making an explicit choice for the sheets of γ will allow fo the establishment of an explicit bijection $E_{\beta\gamma} \leftrightarrow E_\beta \times E_\gamma$.

By definition, the constellation of $\beta\gamma$ is given by an ordering of the edges of $\beta\gamma$ around each ramification point. To this end, each edge will be labeled according to the bijection just described. Then the action of $\pi_1(Z\backslash\{0,1,\infty\}, q)$, for some fixed $q \in \mathbb{I}$, on the edge $e_{\beta\gamma} := (e_\gamma, e_\beta)$ will be determined by considering its action on e_γ and on e_β. In order to determine the action on e_γ, it is necessary to achieve a better understanding of how γ covers $\mathbb{P}^1(\mathbb{C})$. This will be achieved through a specific choice of sheets for γ and a study of how paths in Y permute these sheets. Then, given the way that the dessin of β sits inside $\mathbb{P}^1(\mathbb{C})$, it will be determined how the dessin of β is lifted by an arbitrary Belyĭ map γ. As the bijection $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$ depends on ideas introduced in studying the action of $\pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}, q)$ on $e_{\beta\gamma}$, the latter will be discussed first.

The basic ideas of Sections 4.1. and 4.2. were originally presented in [68], though with less detail. The Corollary to Theorem 4.12 can be found in [38], however the explicit nature of Sections 4.1. and 4.2. allows for a much more explicit formulation of the result than in [38].

1. Covering of $\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$

1.1. Sheets over $\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$

From 1.1., $\mathbb{P}^1(\mathbb{C})$ is the disjoint union of $\mathbb{P}^1(\mathbb{R})$, $\mathbb{H}^+$, and $\mathbb{H}^-$. Moreover, $\mathbb{P}^1(\mathbb{R})$ is the union of the sets

$$\begin{aligned}
[-\infty, 0] &:= \{(u,v,w) \mid w = 0, v \leq 0\},\\
[0,1] &:= \{(u,v,w) \mid w = 0, 0 \leq v \leq 1\}(= \mathbb{I}),\\
[1,\infty] &:= \{(u,v,w) \mid w = 0, v \geq 1\}.
\end{aligned}$$

In addition to the construction given in Section 1.3.3.1., the canonical triangulation can be constructed by lifting via γ the structure on $\mathbb{P}^1(\mathbb{C})$ given by

$$X_0 := \{0, 1, \infty\}, \quad X_1 := \{[-\infty, 0], [0, 1], [1, \infty]\}, \quad X_2 := \{\overline{\mathbb{H}_+}, \overline{\mathbb{H}_-}\}.$$

Lemma 4.1. *If an open set U intersects a set R satisfying $\overline{R^\circ} = \overline{R}$, then $U \cap R$ contains a nonempty open set.*

Proof. If $u \in U \cap R$, then $u \in U \cap \overline{R^\circ}$ and $U \cap R^\circ \neq \emptyset$ is open. □

In the following, many statements will apply to both $\mathbb{H}^+$ and $\mathbb{H}^-$. In order to avoid unnecessary repetition, both cases will be considered simultaneously, as indicated by the use of $\mathbb{H}^\pm$, and similarly for $T^\pm$.

Proposition 4.2.

i. $\mathbb{P}^1(\mathbb{C}) \backslash ([-\infty, 0] \cup [1, \infty])$ *is evenly covered by sets* $\{B_j\}_j$.

ii. $\gamma^{-1}(\overline{\mathbb{H}^\pm}) = \coprod_i \mathcal{T}_i^\pm$, *where for each* i, $\mathcal{T}_i^\pm \approx \overline{\mathbb{H}^\pm}$.

iii. For each $\mathcal{T} \in \{\mathcal{T}_i^+\} \cup \{\mathcal{T}_i^-\}$, $\overline{\mathcal{T}^\circ} = \overline{\mathcal{T}}$. *Hence, if* U *is open and* $U \cap \mathcal{T} \neq \emptyset$, *then* $U \cap \mathcal{T}$ *contains a nonempty open set.*

iv. For each $B \in \{B_j\}_j$, *there is a unique* $\mathcal{T} \in \{\mathcal{T}_i^\pm\}_i$ *so that* $B \cap \mathcal{T} \neq \emptyset$. *Hence, there is a bijection* $\{B_j\}_j \leftrightarrow \Big\{\{\mathcal{T}_i^+, \mathcal{T}_i^-\}\Big\}_i$.

Proof. Let $A := \mathbb{P}^1(\mathbb{C}) \backslash \Big([-\infty, 0] \cup [1, \infty]\Big)$ and $\widetilde{A} := \gamma^{-1}(A)$.

(i) and (ii) follow from Proposition 1.4 by taking $\{B_j\}_j$ and $\{T_i^\pm\}_i$ to be the path components of $\widetilde{A}$ and $\gamma^{-1}\Big(\overline{\mathbb{H}^\pm}\Big)$, respectively.

(iii) As $\overline{\mathbb{H}^\pm} = \overline{(\mathbb{H}^\pm)^\circ}$ and $\mathcal{T}_i^\pm \approx \overline{\mathbb{H}^\pm}$, the result follows by Lemma 4.1.

(iv) Let $T_i^\pm$ be the interior of $\mathcal{T}_i^\pm$. Note that $T_i^\pm \subseteq \widetilde{A}$ as (ii) shows that $\gamma(T_i^\pm) = \mathbb{H}^\pm$. Suppose

$$B_j \cap \mathcal{T}_1^\pm \neq \emptyset, \qquad B_j \cap \mathcal{T}_2^\pm \neq \emptyset.$$

From (iii), $B_j \cap T_i^\pm \neq \emptyset$, $i = 1, 2$. Then because B_j and each $T_i^\pm$ are path connected, $B_j \cup T_1^\pm \cup T_2^\pm$ is path connected. But B_j is a path component of $\widetilde{A}$, implying that

$T_1^{\pm}, T_2^{\pm} \subseteq B_j$. Then the fact that $\gamma|_{B_j}$ is a homeomorphism and $\gamma(T_1)^{\pm} = \gamma(T_2)^{\pm}$ implies that $T_1^{\pm} = T_2^{\pm}$ and $\mathcal{T}_1^{\pm} = \mathcal{T}_2^{\pm}$ by disjointness of path components.

Given $\mathcal{T}_i^{\pm}$, if

$$\mathcal{T}_i^{\pm} \cap B_1 \neq \emptyset, \qquad \mathcal{T}_i^{\pm} \cap B_2 \neq \emptyset,$$

then because $T_i^{\pm}$ is path connected, $B_1 \cup B_2 \cup T_i^{\pm}$ is path connected. As each B_i is a path component of $\widetilde{A}$, $B_1 = B_2$. □

By abuse of terminology, in order to bring the sheets B_i of γ into bijection with $\mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}$, preimages of $(-\infty, 0)$ and $(1, \infty)$ will be included subject.

Definition. *Let B be a path component of $\gamma^{-1}\Big(\mathbb{P}^1(\mathbb{C})\backslash([-\infty, 0] \cup [1, \infty])\Big)$ and let $\{\mathcal{T}^+, \mathcal{T}^-\}$ correspond to B under the bijection in Proposition 4.2(iv). A sheet of γ is*

$$\mathcal{S} := \mathcal{T}^+ \amalg (\mathcal{T}_i^-)^\circ.$$

Convention. *B_j, $\mathcal{T}_i^{\pm}$, and $T_i^{\pm}$ will denote the sets given in the proof of Proposition 4.2. Further, for each i, it will be insisted that $B_i \cap T_i^{\pm}$, in light of Proposition 4.2(iv).*

Lemma 4.3. *Let $a \in \gamma^{-1}\Big(\mathbb{P}^1(\mathbb{R})\Big)$ and let U be an open neighborhood of $\gamma(a)$ which is path connected, locally path connected, and simply connected. Let V be the path component of $\gamma^{-1}(U)$ containing a. Then V intersects exactly one element of $\{T_i^+\}_i$ and exactly one element of $\{T_i^-\}_i$.*

Proof. By Proposition 1.4, $V \approx U$. Consider $V^{\pm} = (\gamma|_V)^{-1}(U \cap \overline{\mathbb{H}^{\pm}})$. As $\gamma(a) \in \partial\mathbb{H}^{\pm}$, so that U being an open neighborhood of $\gamma(a)$ implies $U \cap \mathbb{H}^{\pm} \neq \emptyset$. Then $V = V^+ \cup V^-$ and by Proposition 4.2(ii), V^+, V^- are contained in $\mathcal{T}_i^+, \mathcal{T}_j^-$, respectively. Finally, if $V \cap \mathcal{T} \neq \emptyset$ for some $\mathcal{T} \in \{\mathcal{T}_i^+\}_i \cup \{\mathcal{T}_i^-\}_i$, then $\gamma(V \cap \mathcal{T})$ contains an open neighborhood W by Proposition 4.2(iii) and $\gamma(W) \subseteq \mathbb{H}^+$ or $\gamma(W) \subseteq \mathbb{H}^-$. Because $\gamma|_V$ is a homeomorphism and $\gamma(W) \cap \gamma(T_i^+) \neq \emptyset$ or $\gamma(W) \cap \gamma(T_j^-) \neq \emptyset$, either $W \cap T_i^+ \neq \emptyset$ or $W \cap T_j^- \neq \emptyset$. By disjointness of the $T_i^{\pm}$, $T = T_i^+$ or $T = T_j^-$. □

See also [48, pg. 473-474] for another approach to this proof.

Proposition 4.4. *The boundary of a sheet $\mathcal{S}$ of γ is $\partial T^+ + \partial T^-$, where $+$ denotes the symmetric difference.*

Proof. First, $\partial\mathcal{S} = \partial\mathcal{T}^+ \cup \partial T^- = \partial\mathcal{T}^+ \cup \partial\mathcal{T}^-$. Suppose $a \in \partial\mathcal{T}^+ \cap \partial\mathcal{T}^-$. Let U, V be as in Lemma 4.3. Then given $T \in \bigcup_i \{T_i^+, T_i^-\}$, where $T \neq T^+, T^-$, V does not intersect T. By Proposition 4.2(iii), U does not intersect $\mathcal{T}$ for any $\mathcal{T} \neq \mathcal{T}^+, \mathcal{T}^-$ and U does not intersect $X\backslash\mathcal{S}$. Thus $a \notin \partial\mathcal{S}$ and $\partial\mathcal{S} \subseteq \partial\mathcal{T}^+ + \partial\mathcal{T}^-$.

Now let $a \in \partial\mathcal{T}^+ + \partial\mathcal{T}^-$. As $a \notin \partial\mathcal{T}^\pm$, there exists an open set U with $U \cap \mathcal{T}^\mp = \emptyset$. Then given any open neighborhood V of a, $U \cap V$ contains an open neighborhood V_0 satisfying the hypotheses of Proposition 1.4, for example by taking $V_0 := \gamma^{-1}B_\varepsilon(U) \cap U \cap V$ for sufficiently small $\varepsilon > 0$. As such, V_0 intersects exactly two $\mathcal{T} \in \bigcup_i \{\mathcal{T}_i^+, \mathcal{T}_i^-\}$. One of these is $\mathcal{T}^\pm$. If the other is $\mathcal{T}^\mp$, then U intersects $\mathcal{T}^\mp$, a contradiction. It follows that V_0, hence V, intersects $\mathcal{T} \nsubseteq \mathcal{S}$ and $a \in \overline{X\backslash\mathcal{S}}$. As $a \in \overline{\mathcal{S}}$ by assumption, $a \in \partial\mathcal{S}$. □

Note that if, for example, $(1, \infty) \nsubseteq \partial\mathcal{S}_i$, then the lifting of a loop around 1 does not intersect any sheets other than $\mathcal{S}_i$, hence this lifting begins and ends at the same edge of γ, and the monodromy of γ around this preimage of 1 is trivial. A similar statement holds for $(-\infty, 0) \nsubseteq \partial\mathcal{S}_i$.

Choosing the sheets of γ as above will make the determination of the constellation of $\beta\gamma$ easier. In particular, by respecting the canonical triangulation, this choice will make it easy to identify each sheet with an edge of γ and because $\partial\mathcal{S} \subseteq \gamma^{-1}(\mathbb{P}^1(\mathbb{R}))$, it will be easy to identify when a path in X transitions from one sheet to another.

1.2. Permuting Sheets with Paths

Understanding the relationship among the sheets of γ has been made easier by making an explicit choice for these sheets. In particular, investigation of the action of paths in $\mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}$ on the sheets of γ will provide a method for determining the constellation of $\beta\gamma$ by using paths between the edges of β.

Let $z \in Z$. Because a lifting of a loop in $\pi_1(Z\backslash\{0, 1, \infty\}, z)$ by $\beta\gamma$ is the same as a lifting by β and then by γ, and because a lifting by β usually results in a path, rather than a loop, it is necessary to consider how paths act on the sheets of a covering.

Given an arbitrary path $p \subseteq Y\backslash\{0, 1, \infty\}$, p defines a permutation σ_p on the sheets of γ as follows. Label the sheets of γ by $\{1, \ldots, \deg\gamma\}$ and the points $y_0^{(i)} \in \gamma^{-1}\big(p(0)\big)$

and $y_1^{(i)} \in \gamma^{-1}\big(p(1)\big)$ according to the sheet of γ in which they lie. If $\tilde{p}$ is a lifting of p with $\tilde{p}(0) = y_0^{(i)}$ and $\tilde{p}(1) = y_1^{(j)}$, then define $\sigma_p(i) = j$.

Lemma 4.5. *Let $\gamma : \tilde{R} \to R$ be a covering map. The function*

$$\textit{paths in } R \Big/ \simeq_p \to S_n$$

given by $p \mapsto \sigma_p$ is a homomorphism of groupoids.

Proof. First, let $H : p_1 \simeq_p p_2$ be a path homotopy. By the covering homotopy lemma [51, Cor. 10.6], for any liftings $\tilde{p}_1, \tilde{p}_2$ of p_1, p_2, respectively, $\tilde{p}_1(1) = \tilde{p}_2(1)$, and the function is well-defined.

Consider the path $p_1 * p_2$ and let $y_0 := p_1(0)$, $y_1 := p_1(1) = p_2(0)$, and $y_2 := p_2(1)$. Let the points lying over y_i be $y_i^{(j)}$, according to some labeling. Let $y_0^{(i)}$ be a point lying over y_0 and let $\tilde{p}_1$ be the lifting of p_1 with $\tilde{p}_1(0) = y_0^{(i)}$. Then there is a unique lifting $\tilde{p}_2$ of p_2 with $\tilde{p}_2(0) = \tilde{p}_1(1)$. Setting $j := i^{\sigma_{p_1}}$ and $k := j^{\sigma_{p_2}}$,

$$\tilde{p}_2(0) = \tilde{p}_1(1) = y_1^{(j)} \qquad \text{and} \qquad \tilde{p}_2(1) = y_2^{(k)}.$$

As $k = j^{\sigma_{p_2}} = i^{\sigma_{p_1}\sigma_{p_2}}$ and $(p_1 * p_2)\tilde{}(1) = y_2^{(k)}$, where $(p_1 * p_2)\tilde{}$ is the lifting of $p_1 * p_2$ beginning at $y_0^{(i)}$, it follows that $i^{\sigma_{p_1 * p_2}} = k = i^{\sigma_{p_1}\sigma_{p_2}}$. Because i was arbitrary, $\sigma_{p_1 * p_2} = \sigma_{p_1}\sigma_{p_2}$. □

By relating the sheets of γ with the edges of γ, it will be possible to determine the permutation defined by a path p from the monodromy of γ. Because the sheets of γ have boundary contained in $\gamma^{-1}(\mathbb{P}^1(\mathbb{R}))$, the focus will be on places where the paths in Y cross $\mathbb{P}^1(\mathbb{R})$.

Convention.

$$\begin{aligned}
\mathcal{R}_{-1/2} &:= \mathbb{P}^1(\mathbb{C}) \backslash [0, \infty], \\
\mathcal{R}_{1/2} &:= \mathbb{P}^1(\mathbb{C}) \backslash ([-\infty, 0] \cup [1, \infty]), \\
\mathcal{R}_{3/2} &:= \mathbb{P}^1(\mathbb{C}) \backslash [-\infty, 1].
\end{aligned}$$

Note that the subscript of $\mathcal{R}$ indicates the unique real point of $\mathcal{R}$ which lies on either of $e^{2\pi i t}/2$ or $1 - e^{2\pi i t}/2$. Further, each $\mathcal{R}.$ is star-shaped.

Lemma 4.6. *Let $p \subseteq \mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$ be a path. Then p can be decomposed into paths $\{p_i\}_i$ with $p_i \subseteq \mathbb{P}^1(\mathbb{C})\backslash[-\infty,1]$, $\mathbb{P}^1(\mathbb{C})\backslash[0,\infty]$, or $\mathbb{P}^1(\mathbb{C})\backslash([-\infty,0]\cup[1,\infty])$.*

Proof. Suppose p is contained in more than one such region. Let $p(0) \in \mathcal{R}$ and let $t_1 = \inf\{t \in \mathbb{I} \mid p(t) \notin \mathcal{R}\}$. Then $p(t_1) \in \mathbb{P}^1(\mathbb{R})\backslash\{0,1,\infty\}$. Because $p([0,t_1])$ is connected, while $\mathbb{P}^1(\mathbb{R})\backslash\{0,1,\infty\}$ is not, $p([0,t_1]) \nsubseteq \mathbb{P}^1(\mathbb{R})\backslash\{0,1,\infty\}$ and there exists $t_0 \in [0,t_1]$ so that $p(t_0) \in \mathbb{P}^1(\mathbb{C})\backslash\mathbb{P}^1(\mathbb{R})$. Then $p|_{[0,t_0]} \subseteq \mathcal{R}$. □

From Proposition 4.2(i), each sheet can be uniquely associated with the edge of γ which it contains. As such, the sheets containing $\tilde{p}(0)$ and $\tilde{p}(1)$ will be determined by constructing from p a loop $p^{\circlearrowleft}$ so that $p(0)$ and $p(1)$ lie in the same sheet as $p^{\circlearrowleft}(0)$ and $p^{\circlearrowleft}(1)$, respectively, and each lifting of $p^{\circlearrowleft}$ by γ is a path between edges of γ. As $\partial\mathcal{S} \subseteq \gamma^{-1}\Big(\mathbb{P}^1(\mathbb{R})\backslash([0,1]\cup\{\infty\})\Big)$ and $\gamma^{-1}([0,1])$ constitutes the edges of γ, any extension of p should not cross $\mathbb{P}^1(\mathbb{R})\backslash\{0,1,\infty\}$.

Construction 4.7. *Let p be a path in $\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$ which is contained in one of the star-shaped regions $\mathcal{R}_{-1/2}, \mathcal{R}_{1/2}, \mathcal{R}_{3/2}$. Being star-shaped, each $\mathcal{R}_{a_0}$ is contractible [61, pg. 29], hence simply connected, and there is a unique path class connecting $p(0)$ and $p(1)$ [41, Cor. 4.3, Exer. 4.10]. Thus, any path $p \in \mathcal{R}_{a_0}$ is path homotopic to*

$$q_{a_0} := \Big((1-t)p(0) + ta_0\Big)\cdot\Big((1-t)a_0 + tp(1)\Big).$$

As such, let $s := \operatorname{sgn}\Big(\operatorname{Re} p(i)\Big)$ and let

$$\alpha_i(t) := \begin{cases} (1-t)p(i) + \frac{1}{2}t & \text{if } p(i) \notin \mathcal{R}_{1/2} \\ \frac{1/2+p(i)}{2} + \frac{1/2-p(i)}{2}e^{s\pi i t} & \text{if } p(i) \notin \mathcal{R}_{1/2} \end{cases},$$

$$p^{\circlearrowleft} := \alpha_0^{-1} q_{a_0} \alpha_1.$$

The function $\alpha_i(t)$ when $p(i) \notin \mathcal{R}_{1/2}$ is a half-circle from $p(i)$ to $1/2$ which passes through $\mathbb{H}^+$.

Lemma 4.8. *The function*

$$\cdot^{\circlearrowleft} : \text{paths in } \mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}\big/\simeq_p \;\to \pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}, q)$$

$$p \mapsto p^{\circlearrowleft}$$

is a homomorphism of groupoids and $\sigma_p = \sigma_{p^{\circlearrowleft}}$*.*

Proof. Let $H : p_1 \simeq_p p_2$ the paths $\alpha_i^{(j)}$ used to form $p_j^{\circlearrowleft}$ from p_j depend only on the endpoints of p_j, the path homotopy $\mathrm{id} \cdot H \cdot \mathrm{id}$ shows that $p_1^{\circlearrowleft} \simeq_p p_2^{\circlearrowleft}$ and $\cdot^{\circlearrowleft}$ is well-defined. Suppose now that $p_1, p_2 \subseteq \mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}$ are paths with $p_1(1) = p_2(0)$. As $\alpha_1^{(1)} = \alpha_0^{(2)}$,

$$p_1^{\circlearrowleft} p_2^{\circlearrowleft} = (\alpha_0^{(1)})^{-1} q_1 \alpha_1^{(1)} (\alpha_0^{(2)})^{-1} q_2 \alpha_1^{(2)} \simeq_p (\alpha_0^{(1)})^{-1} q_1 q_2 \alpha_1^{(2)}.$$

But $p_j \simeq_p q_j$, so that $p_1 * p_2 \simeq_p q_1 q_2$ and

$$(\alpha_0^{(1)})^{-1} q_1 q_2 \alpha_1^{(2)} \simeq_p (\alpha_0^{(1)})^{-1} p_1 * p_2 \alpha_1^{(2)},$$

noting that $\alpha_0^{(1)}, \alpha_1^{(2)}$ are also the paths used to form $(p_1 * p_2)^{\circlearrowleft}$ from $p_1 * p_2$. Thus, $\cdot^{\circlearrowleft}$ is a homomorphism.

To see that $\sigma_p = \sigma_{p^{\circlearrowleft}}$, suppose first that $\gamma p(i) \in \mathbb{P}^1(\mathbb{C})\backslash([-\infty, 0] \cup [1, \infty])$. Then $\gamma \alpha_i \subseteq \mathbb{P}^1(\mathbb{C})\backslash([-\infty, 0] \cup [1, \infty])$. As α_i is a path from $p(i)$ to $1/2$, it must lie in the path component of $\gamma^{-1}\Big(\mathbb{P}^1(\mathbb{C})\backslash([-\infty, 0] \cup [1, \infty])\Big)$ containing $p(i)$, hence lies in the same sheet as $p(i)$.

On the other hand, if $\gamma p(i) \in (-\infty, 0) \cup (1, \infty)$, then $\gamma \alpha_i \subseteq \overline{\mathbb{H}^+}$. Because α_i is a path in $\gamma^{-1}(\overline{\mathbb{H}^+})$ from $p(i)$ to $1/2$, it must lie in the path component of $\gamma^{-1}(\overline{\mathbb{H}^+})$ containing $p(i)$, hence lies in the same $\mathcal{T}^+$ as $p(i)$, and lies in the same sheet as $p(i)$.

Thus, for $i = 0, 1$, $p^{\circlearrowleft}(i)$ lies in the same sheet of γ as $p(i)$ and defines the same permutation as $p(i)$. □

Convention. *Let* a*,* b*, and* c *be simple closed curves having winding number one around* 0*,* 1*, and* ∞*, respectively, and winding number zero around* 1 *and* ∞*,* 0 *and* ∞*, and* 0 *and* 1*, respectively.*

The endpoints of $p^{\circlearrowleft}$ are $1/2$, hence $p^{\circlearrowleft} \in \pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}, 1/2)$ and $\sigma_{p^{\circlearrowleft}}$ is just the image of $p^{\circlearrowleft}$ under the monodromy representation of $\pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}, 1/2)$. To this end, the path-homotopy class of $p^{\circlearrowleft}$ in $\pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0, 1, \infty\}, q)$ is as follows (see Figure 4.2).

1. If $p \subseteq \mathcal{R}_{1/2}$, then $p^{\circlearrowleft} \simeq_p 1$.

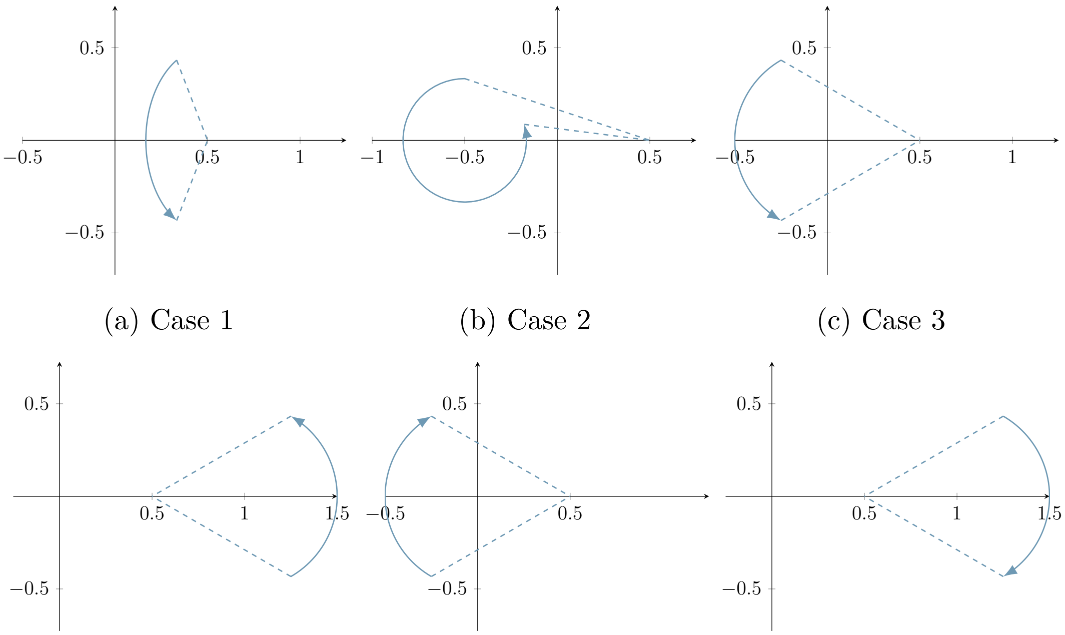
2. If either $p(0), p(1) \in \overline{\mathbb{H}^+}$ or $p(0), p(1) \in \mathbb{H}^-$ and either $p \subseteq \mathcal{R}_{-1/2}$ or $p \subseteq \mathcal{R}_{3/2}$, then $p^{\circlearrowleft} \simeq_p 1$.

3. If $p(0) \in \overline{\mathbb{H}^+}$, $p(1) \in \mathbb{H}^-$, and $p \subseteq \mathcal{R}_{-1/2}$, then $p^{\circlearrowleft} \simeq_p a$.

4. If $p(0) \in \overline{\mathbb{H}^+}$, $p(1) \in \mathbb{H}^-$, and $p \subseteq \mathcal{R}_{3/2}$, then $p^{\circlearrowleft} \simeq_p b^{-1}$.

5. If $p(0) \in \mathbb{H}^-$, $p(1) \in \overline{\mathbb{H}^+}$, and $p \subseteq \mathcal{R}_{-1/2}$, then $p^{\circlearrowleft} \simeq_p a^{-1}$.

6. If $p(0) \in \mathbb{H}^-$, $p(1) \in \overline{\mathbb{H}^+}$, and $p \subseteq \mathcal{R}_{3/2}$, then $p^{\circlearrowleft} \simeq_p b$.



(a) Case 1

(b) Case 2

(c) Case 3

(d) Case 4

(e) Case 5

(f) Case 6

Figure 4.2.: The path-homotopy classes of extensions of basic paths

Noteworthy cases of using the above to determine the action of paths crossing $\mathbb{P}^1(\mathbb{R})$ multiple times are the loops around ∞. If p can be decomposed into

7. a path satisfying (3) followed by a path satisfying (6), then $p^{\circlearrowleft} \simeq ab$.

8. a path satisfying (4) followed by a path satisfying (5), then $p^{\circlearrowleft} \simeq b^{-1}a^{-1}$.

Now let p be an arbitrary path which is decomposed into $p_1, \ldots, p_r$, with each $p_i \subseteq \mathbb{P}^1(\mathbb{R})$ or crossing $\mathbb{P}^1(\mathbb{R})$ at most once. Let the monodromy representation of γ be ρ_γ. Then σ_p is determined by Lemma 4.5 and Lemma 4.8:

$$\sigma_{p_1 * p_2} = \sigma_{(p_1 * p_2)^{\circlearrowleft}} = \sigma_{p_1^{\circlearrowleft} p_2^{\circlearrowleft}} = \rho_\gamma(p_1^{\circlearrowleft} p_2^{\circlearrowleft}) = \rho_\gamma(p_1^{\circlearrowleft})\rho_\gamma(p_2^{\circlearrowleft}) = \sigma_{p_1^{\circlearrowleft}}\sigma_{p_2^{\circlearrowleft}} = \sigma_{p_1}\sigma_{p_2}.$$

2. Action of Loops

Before determining the constellation of $\beta\gamma$, each edge of the constellation must be labelled. To do so, each edge will be associated with a sheet of γ and an edge of β. In the determination of the extending pattern, the basepoint q will be taken to be $1/2 \in \mathbb{I}$ in order to establish the bijection given below. This choice is arbitrary and could be replaced by any point in $(0, 1)$ throughout.

2.1. The Real Case

Lemma 4.9. *Let β be a Belyĭ map of genus zero defined over $\mathbb{R}$ and let $p(t)$ be an edge of β. Either $p(\mathbb{I}^\circ) \cap \mathbb{P}^1(\mathbb{R}) \neq \emptyset$ or $p(\mathbb{I}^\circ) \subseteq \mathbb{P}^1(\mathbb{R})$.*

Proof. By Lemma 1.6, replace p by a holomorphic function which agrees with p on $\mathbb{I}^\circ$, p is holomorphic on $\mathbb{E}$ and $\beta p(z) = z$ for $z \in \mathbb{E} \cap \mathbb{P}^1(\mathbb{R})$. Let $a \in W := p^{-1}(\mathbb{P}^1(\mathbb{R})) \cap \mathbb{I}^\circ$. Note that $\mathbb{P}^1(\mathbb{R})$ is closed in $\mathbb{P}^1(\mathbb{C})$, implying that W is closed in $\mathbb{I}^\circ$ by the continuity of p. As $\beta p(a) = a \notin \{0, 1, \infty\}$, β is unramified at $p(a)$ and there is an open neighborhood U of $p(a)$ on which β is injective by the injectivity lemma [49, pg. 282]. But β is defined over $\mathbb{R}$, so that if $\beta p(b) \in \mathbb{P}^1(\mathbb{R})$ and $p(b) \in U$, then $\beta p(b) = \beta\overline{p(b)}$ implies $p(b) = \overline{p(b)}$ and $p(b) \in \mathbb{R}$.

Thus, if $b \in p^{-1}(U) \cap \mathbb{I}^\circ$, then because $\beta p = \text{id}$ on $\mathbb{I}^\circ$, $\beta p(b) = b \in \mathbb{P}^1(\mathbb{R})$, implying that $p(b) \in \mathbb{P}^1(\mathbb{R})$, so that $b \in W$. This implies that $p^{-1}(U) \cap \mathbb{I}^\circ$ is an open neighborhood of a contained in W and W is open. Being an open and closed subset of the connected space $\mathbb{I}^\circ$, either $W \neq \emptyset$ or $W = \mathbb{I}^\circ$ and the result follows. □

What this says is that if β is defined over $\mathbb{R}$, the interior of each edge of β is either contained in $\mathbb{P}^1(\mathbb{R})$ or is contained in either $\mathbb{H}_+$ or $\mathbb{H}_-$. In particular, if e_β is an edge of β, then $\gamma\Big(e_\beta(\mathbb{I}^\circ)\Big)$ is contained in a unique sheet of γ.

As $\beta(\{0,1,\infty\}) \subseteq \{0,1,\infty\}$, $\{0,1,\infty\} \cap \beta^{-1}(0,1) = \emptyset$ and each of the $\deg\gamma$ disjoint sheets over Y contains a subset Σ so that $\beta : \Sigma \leftrightarrow (0,1)$. Moreover, there is a bijection $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$ as follows. Labeling each sheet of γ over Y by $\{1,\ldots,\deg\gamma\}$ and each edge of β by $\{1,\ldots,\deg\beta\}$, each edge of $\beta\gamma$ can be identified by the sheet of γ in which it lies and the edge of β which it lies over. Each sheet of γ contains a unique edge of γ, resulting in a bijection $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$.

2.2. The Complex Case

In the case that β is not defined over $\mathbb{R}$, the edges of β are no longer guaranteed to be contained in either $\overline{\mathbb{H}_+}$ or $\mathbb{H}_-$. As a result, the edges of $\beta\gamma$ are not guarenteed to lie in a single sheet of γ and a choice must be made as to how to label the edges which lie in multiple sheets. To this end, each edge e of $\beta\gamma$ will be identified with the sheet containing $e(1/2)$. Note that $\beta e(1/2) = 1/2$, so that $e(1/2) \notin \{0,1,\infty\}$ and $e(1/2)$ lies in a unique sheet (see Figure 4.3).

Lemma 4.10. *Given an edge e of $\beta\gamma$, let $\mathcal{S}_e$ be the sheet of γ such that $e(1/2) \in \mathcal{S}$. There is a bijection $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$ given by*

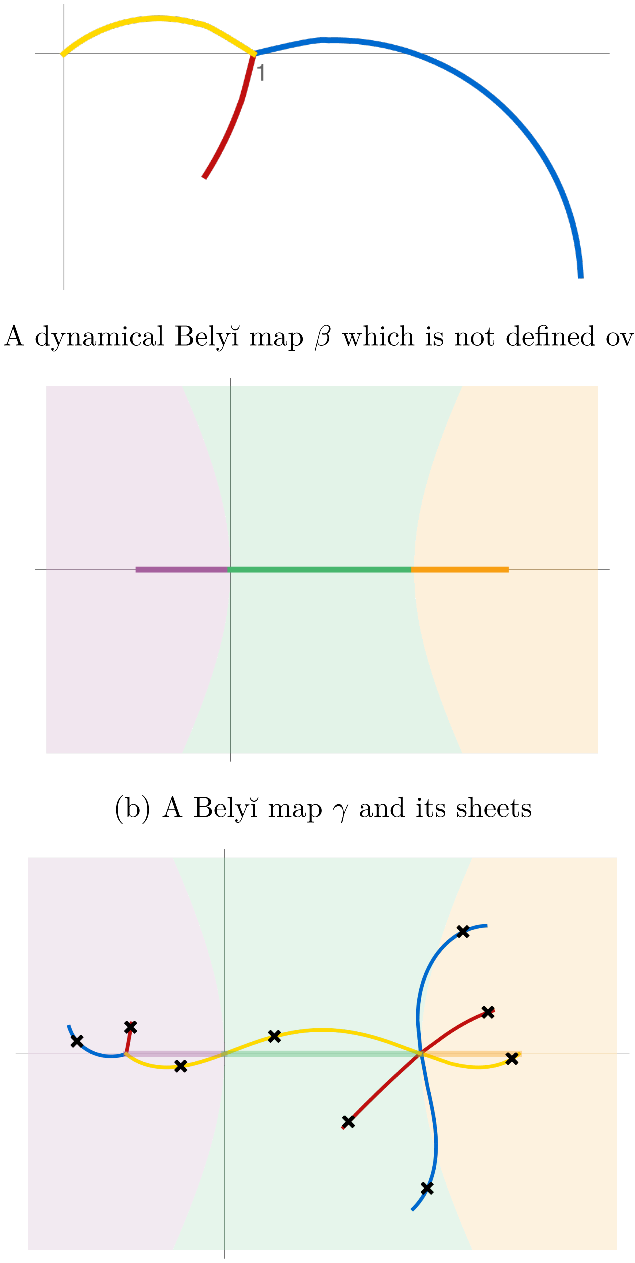
$$e \longleftrightarrow \Big(e_\gamma \text{ such that } e_\gamma \in \mathcal{S}_e, \gamma(e)\Big). \tag{4.1}$$

In Figure 4.3, the preimage of $\mathbb{P}^1(\mathbb{R})\backslash[0,1]$ serves to demarcate the sheets of γ according the the edge of γ which they contain. Each edge is labeled by the sheet containing its midpoint and the unique edge of β which it lies over.

2.3. The Action of a Loop

Recall the situation under consideration: $X \xrightarrow{\gamma} Y := \mathbb{P}^1(\mathbb{C}) \xrightarrow{\beta} Z := \mathbb{P}^1(\mathbb{C})$. Using the given bijection, it is possible to consider the effect of a loop on each component of an edge $e := (e_\gamma, e_\beta) \in E_\gamma \times E_\beta$.

Let λ be a loop with basepoint $1/2$ in $Z\backslash\{0,1,\infty\}$. Let the monodromy representation of β, respectively γ, be denoted ρ_β, respectively ρ_γ. The action of λ on the



(a) A dynamical Belyĭ map β which is not defined over $\mathbb{R}$

(b) A Belyĭ map γ and its sheets

(c) The Belyĭ map $\beta\gamma$; **x**'s indicate preimages of $1/2$

Figure 4.3.: A composition of Belyĭ maps demonstrating the bijection $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$

E_β component of an edge $e \in E_{\beta\gamma}$ is as expected. Let λ_X be a lifting of λ by $\beta\gamma$ with $\lambda_X(0) = e(1/2)$. As $\beta(\gamma\lambda_X) = \lambda$, $\lambda_Y := \gamma\lambda_X$ is a lifting of λ by β. Moreover, $\lambda_Y(0) = e_\beta(1/2)$ as e lies over e_β, hence $\gamma\lambda_X(1) = \lambda_Y(1) \in e_\beta \cdot \rho_\beta(\lambda)$. Thus, $\lambda_X(1)$ lies over a point in $e_\beta \cdot \rho_\beta(\lambda)$ and the E_β component of e^λ is given by $e_\beta \cdot \rho_\beta(\lambda)$ (see Figure 4.4).

It remains to determine the action of λ on the E_γ-component of $e \in E_{\beta\gamma}$. Under the established bijection $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$, the E_γ component of e^λ is the sheet of γ in which $\lambda_X(1)$ lies. To determine this sheet, form the loop $\lambda_Y^{\circlearrowleft}$ from λ_Y according to Construction 4.7. Consider the lifting p of $\lambda_Y^{\circlearrowleft}$ with $p(0) \in e_\gamma$. As $\lambda_Y^{\circlearrowleft} \in \pi_1(Y\backslash\{0,1,\infty\}, 1/2)$, p is a path between edges of γ and $p(1)$ lies on the edge $e_\gamma \cdot \rho_\gamma(\lambda_Y^{\circlearrowleft})$. By Lemma 4.8, $\lambda_X(1)$, which lies over $\lambda_Y(1)$, is in the same sheet as $p(1)$, which lies over $\lambda_Y^{\circlearrowleft}(1)$. Thus, the E_γ component of e^λ is the sheet containing $p(1)$, which is given by $e_\gamma \cdot \rho_\gamma(\lambda_Y^{\circlearrowleft})$.

Putting the action on each component together,

$$(e_\gamma, e_\beta)^\lambda = \Big(e_\gamma \cdot \rho_\gamma(p_Y^{\circlearrowleft}), e_\beta \cdot \rho_\beta(\lambda)\Big).$$

2.4. Independent of γ

In fact, the path-homotopy class of $\lambda_Y^{\circlearrowleft}$ is independent of γ, as λ_Y is simply a lifting of λ by β. However, $\lambda_Y^{\circlearrowleft}$ does depend on the edge e_β of β containing $\lambda_Y(0)$. Thus, a loop $\lambda \subseteq Z\backslash\{0,1,\infty\}$ with basepoint $1/2$ defines a function $f_\lambda : E_\beta \to \pi_1(Y\backslash\{0,1,\infty\}, 1/2)$ by

$$f_\lambda : e_\beta \mapsto \begin{array}{c} \lambda_Y^{\circlearrowleft} \in \pi_1(Y\backslash\{0,1,\infty\}, 1/2) \text{ such that} \\ \lambda_Y(0) = e_\beta(1/2) \text{ and } \beta\lambda_Y = \lambda. \end{array} \tag{4.2}$$

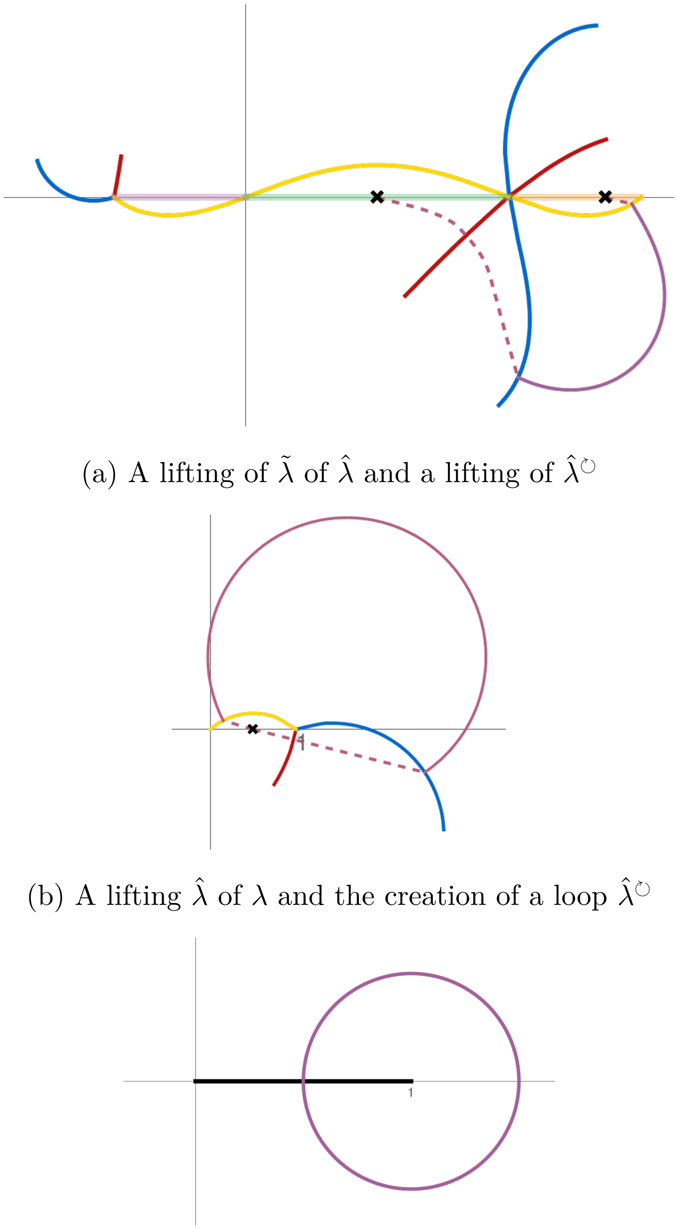
With such a function, the action of λ on (e_β, e_γ) takes the form

$$(e_\gamma, e_\beta)^\lambda = \Big(e_\gamma^{f_\lambda(e_\beta)}, e_\beta^\lambda\Big), \tag{4.3}$$

where the actions are those of the monodromy representations.

2.5. Lifting a Homotopy

To this point, the consideration has been the action of a particular loop. However, for dynamical Belyĭ maps, this action is constant on path-homotopy classes.



(a) A lifting of $\tilde{\lambda}$ of $\hat{\lambda}$ and a lifting of $\hat{\lambda}^{\circlearrowright}$

(b) A lifting $\hat{\lambda}$ of λ and the creation of a loop $\hat{\lambda}^{\circlearrowright}$

(c) A loop λ around 1

Figure 4.4.: Lifting a loop in Z by $\beta\gamma$ determines a path between edges of γ

Lemma 4.11. *Let β be a dynamical Belyĭ map. If $p_1 \simeq_p p_2$ in $Z\backslash\{0,1,\infty\}$ and $\tilde{p}_1, \tilde{p}_2$ are liftings to Y by β with $\tilde{p}_1(0) = \tilde{p}_2(0)$, then $\tilde{p}_1 \simeq_p \tilde{p}_2$ in $Y\backslash\{0,1,\infty\}$.*

Proof. Let $H : p_1 \simeq_p p_2$. Being a Belyĭ map, $\Big(\beta, \mathbb{P}^1(\mathbb{C})\backslash\beta^{-1}(\{0,1,\infty\})\Big)$ is a covering space of $\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$, and being dynamical, $\{0,1,\infty\} \subseteq \beta^{-1}(\{0,1,\infty\})$. By the Covering Homotopy Lemma [51, Corollary 10.6], there exists a lifting $\tilde{H} : \tilde{p}_1 \simeq_p \tilde{p}_2$ of H by β to $\mathbb{P}^1(\mathbb{C})\backslash\beta^{-1}(\{0,1,\infty\})$. As $\tilde{H} \subseteq \mathbb{P}^1(\mathbb{C})\backslash\beta^{-1}(\{0,1,\infty\}) \subseteq \mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$, $\tilde{H}$ is a path homotopy of $\tilde{p}_1, \tilde{p}_2$ in $\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$ as well. □

Then by Lemma 4.8, $\lambda \mapsto \lambda_Y^{\circlearrowleft}$ is a function

$$\pi_1(Z\backslash\{0,1,\infty\}, 1/2) \to \pi_1(Y\backslash\{0,1,\infty\}, 1/2)$$

and $\lambda \mapsto f_\lambda$ is a function $\pi_1(Z\backslash\{0,1,\infty\}, 1/2) \to \operatorname{Fun}\Big(E_\beta, \pi_1(Y\backslash\{0,1,\infty\}, 1/2)\Big)$.

3. Monodromy as a Wreath Product

Let $\pi_1^Z := \pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}, 1/2)$ and $\pi_1^Y := \pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}, 1/2)$. Define the action of $\operatorname{Mon}\beta$ on $\operatorname{Fun}(E_\beta, F_2)$ by $f^\sigma(e_\beta) = f(e_\beta^{\sigma^{-1}})$.

Theorem 4.12. *Let β be a dynamical Belyĭ map, let ρ_β be its monodromy representation, let $\tau_\lambda = \rho_\beta(\lambda)$, and let f_λ be defined as in* (4.2)*.*

1. *For $\lambda \in \pi_1^Z$, define*

   $$\varphi(\lambda) = f_\lambda \rtimes \tau_\lambda.$$

   Then φ is a homomorphism

   $$\varphi : \pi_1^Z \to \pi_1^Y \wr_{E_\beta} \operatorname{Mon}\beta = \operatorname{Fun}(E_\beta, \pi_1^Y) \rtimes \operatorname{Mon}\beta.$$

2. *For any Belyĭ map γ, let ρ_γ be its monodromy representation and define*

   $$\rho_{\gamma *} : \operatorname{Fun}(E_\beta, F_2) \to \operatorname{Fun}(E_\beta, \operatorname{Mon}\gamma)$$

   by $\rho_{\gamma *}(f) = \rho_\gamma \circ f$. Then $\varphi_\gamma := (\rho_{\gamma *} \rtimes \mathrm{id}) \circ \varphi$ factors through $\rho_{\beta\gamma}$ and

   $$w_\gamma := \operatorname{Mon}\beta\gamma \hookrightarrow \operatorname{Mon}\gamma \wr_{E_\beta} \operatorname{Mon}\beta,$$

   defined by $w_\gamma(\tau_\lambda) := \varphi_\gamma(\lambda)$, is an injection:

$$\begin{tikzcd}
F_2 \arrow[r, "\varphi"] \arrow[drr, "\rho_{\beta\gamma}"'] & F_2 \wr_{E_\beta} \operatorname{Mon}\beta \arrow[r, "\rho_{\gamma *}\rtimes \mathrm{id}"] & \operatorname{Mon}\gamma \wr_{E_\beta} \operatorname{Mon}\beta \leq S_{E_\gamma \times E_\beta} \\
 & & \operatorname{Mon}\beta\gamma \leq S_{E_\gamma \times E_\beta} \arrow[u, hook, "w_\gamma"']
\end{tikzcd}.$$

Proof. Let $\lambda_1, \lambda_2 \in \pi_1^Z$. Then $\varphi(\lambda_1 * \lambda_2) = (f_{\lambda_1 * \lambda_2}, \tau_{\lambda_1 * \lambda_2})$. On the other hand, because ρ_β is a homomorphism,

$$\varphi(\lambda_1)\varphi(\lambda_2) = (f_{\lambda_1}, \tau_{\lambda_1})(f_{\lambda_2}, \tau_{\lambda_2}) = \Big(f_{\lambda_1} \cdot f_{\lambda_2}^{\tau_{\lambda_1}^{-1}}, \tau_{\lambda_1 * \lambda_2}\Big).$$

It remains to show that $f_{\lambda_1 * \lambda_2} = f_{\lambda_1} \cdot f_{\lambda_2}^{\tau_{\lambda_1}^{-1}}$. Let $e_\beta \in E_\beta$. Then λ_1 lifts to $\tilde{\lambda}_1$ with $\tilde{\lambda}_1 = e_\beta(1/2)$ and λ_2 lifts to $\tilde{\lambda}_2$ with $\tilde{\lambda}_2(0) = \tilde{\lambda}_1(1)$. By uniqueness of liftings, $\lambda_1 * \lambda_2$ lifts to $\tilde{\lambda}_1\tilde{\lambda}_2$. As $e_\beta^{\tau_{\lambda_1}}$ is the edge e with $e(1/2) = \tilde{\lambda}_1(1) = \tilde{\lambda}_2(0)$, $\tilde{\lambda}_2$ is the lifting with $\tilde{\lambda}_2(0) \in e_\beta^{\tau_{\lambda_1}}$. Therefore, $f_{\lambda_2}(e_\beta^{\tau_{\lambda_1}})$ is the loop $\tilde{\lambda}_2^{\circlearrowleft}$ obtained from the lifting $\tilde{\lambda}_2$ of λ and

$$\begin{aligned}
(f_{\lambda_1} \cdot f_{\lambda_2}^{\tau_{\lambda_1}^{-1}})(e_\beta) &= f_{\lambda_1}(e_\beta) f_{\lambda_2}(e_\beta^{\tau_{\lambda_1}}) \\
&= \tilde{\lambda}_1^{\circlearrowleft}\tilde{\lambda}_2^{\circlearrowleft} \\
&\simeq_p (\tilde{\lambda}_1\tilde{\lambda}_2)^{\circlearrowleft} \qquad \text{(by Lemma 4.8)} \\
&= f_{\lambda_1 * \lambda_2}(e_\beta).
\end{aligned}$$

Because e_β was arbitrary, $f_{\lambda_1} \cdot f_{\lambda_2}^{\tau_{\lambda_1}} = f_{\lambda_1 * \lambda_2}$ and φ is a homomorphism.

Next, for $e_\beta \in E_\beta$ and $(f_j, \tau_j) \in \operatorname{Fun}(E_\beta, \pi_1^Y) \rtimes \operatorname{Mon}\beta$, $j = 1, 2$,

$$\rho_{\gamma *}(f_1 f_2)(e_\beta) = \rho_\gamma\Big(f_1 f_2(e_\beta)\Big) = \rho_\gamma\Big(f_1(e_\beta)\Big)\rho_\gamma\Big(f_2(e_\beta)\Big) = \Big(\rho_{\gamma *}(f_1)\rho_{\gamma *}(f_2)\Big)(e_\beta)$$

shows that $\rho_{\gamma *}$ is a homomorphism. Finally, φ_γ is a homomorphism as follows. Let $\rho := \rho_{\gamma *} \rtimes \mathrm{id}$. Then for $(f_j, \tau_j) \in \operatorname{Fun}(E_\beta, F_2) \rtimes \operatorname{Mon}\beta$, $j = 1, 2$,

$$\rho\Big((f_1, \tau_1), (f_2, \tau_2)\Big) = \rho\Big(f_1 f_2^{\tau_1^{-1}}, \tau_1\tau_2\Big) = \Big(\rho_{\gamma *}(f_1)\rho_{\gamma *}(f_2^{\tau_1^{-1}}), \tau_1\tau_2\Big).$$

On the other hand,

$$\rho(f_1, \tau_1)\rho(f_2, \tau_2) = \Big(\rho_{\gamma *}(f_1), \tau_1\Big)\Big(\rho_{\gamma *}(f_2), \tau_2\Big) = \Big(\rho_{\gamma *}(f_1)\rho_{\gamma *}(f_2)^{\tau_1}, \tau_1\tau_2\Big).$$

But $\rho_{\gamma*}(f_2)^{\tau_1^{-1}}(e_\beta) = \rho_{\gamma*}\Big(f_2(e_\beta^{\tau_1^{-1}})\Big) = \rho_{\gamma*}\Big(f_2^{\tau_1^{-1}}(e_\beta)\Big)$, so that

$$\rho(f_1,\tau_1)\rho(f_2,\tau_2) = \Big(\rho_{\gamma*}(f_1)\rho_{\gamma*}(f_2^{\tau_1^{-1}}), \tau_1\tau_2\Big)$$

and ρ is a homomorphism. As a composition of homomorphisms, φ_γ is thus also a homomorphism.

Upon showing that $\ker\rho_{\beta\gamma} = \ker\varphi_\gamma$, it follows from the first isomorphism theorem that $\operatorname{Mon}\beta\gamma \approx \varphi_\gamma(\pi_1^Z)$. Letting w_γ denote this isomorphism, this would imply $w_\gamma(\tau_\lambda) = \varphi_\gamma(\lambda)$, completing the proof.

If $\lambda \in \ker\rho_{\beta\gamma}$, then for all $(e_\gamma, e_\beta) \in E_\gamma \times E_\beta$, $(e_\gamma, e_\beta)^\lambda = (e_\gamma, e_\beta)$ and $\tau_\lambda = 1$. Moreover, from (2.3.), $e_\gamma \cdot \rho_\gamma f_\lambda(e_\beta) = e_\gamma$ for all e_β, e_γ. Thus, $\varphi_\gamma(\lambda) = 1$. On the other hand, if $\varphi_\gamma(\lambda) = 1$, then $\tau_\lambda = 1$ and $\rho_\gamma f_\lambda(e_\beta) = 1$ for all e_β, so that by (2.3.), $(e_\gamma, e_\beta)^\lambda = (e_\gamma, e_\beta)$. That is, $\lambda \in \ker\rho_{\beta\gamma}$. □

Corollary. *The monodromy group* $\operatorname{Mon}\beta\gamma$ *of the composition of a dynamical Belyĭ map* β *and a Belyĭ map* γ *is isomorphic to a subgroup of the wreath product* $\operatorname{Mon}\gamma \wr_{E_\beta} \operatorname{Mon}\beta$. *Moreover, this isomorphism is given by*

$$\begin{aligned} \operatorname{Mon}\beta\gamma &\;\rightarrow\; \varphi_\gamma(\pi_1^Z) \leq \operatorname{Mon}\gamma \wr_{E_\beta} \operatorname{Mon}\beta \\ \rho_{\beta\gamma}(\lambda) &\;\mapsto\; \Big(\rho_{\gamma*}(f_\lambda), \rho_\beta(\lambda)\Big). \end{aligned}$$

3.1. The Extending Pattern

As a, b generate π_1^Z, $\varphi(\pi_1^Z)$ is generated by (f_a, τ_a) and (f_b, τ_b). Form the isomorphism in the corollary, $\rho_{\beta\gamma}(g_i)$ can be determined from ρ_γ, f_a, f_b, and τ_a, τ_b. As such, the additional information necessary to determine $\Delta_{\beta\gamma}$ from Δ_β and Δ_γ is f_a, f_b.

Definition. *The extending pattern of* β *at* j, $j = 0, 1$, *is the function* $f_j : E_\beta \to \pi_1(\mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}, 1/2)$ *defined as follows. Given* a, b, *for each* $e_\beta \in E_\beta$, *lift* a, b, *by* β *to* $\tilde{a}, \tilde{b}$ *so that* $\tilde{a}(0) = \tilde{b}(0) = e_\beta(1/2)$. *Form* $a^{\circlearrowleft}, b^{\circlearrowleft}$ *from* $\tilde{a}, \tilde{b}$ *as in Construction 4.7. Then* $f_0(e_\beta) = [a^{\circlearrowleft}]$ *and* $f_1(e_\beta) = [b^{\circlearrowleft}]$, *where* $[\lambda]$ *is the path-homotopy class of* λ.

The extending pattern of β *is the pair of functions* (f_0, f_1).

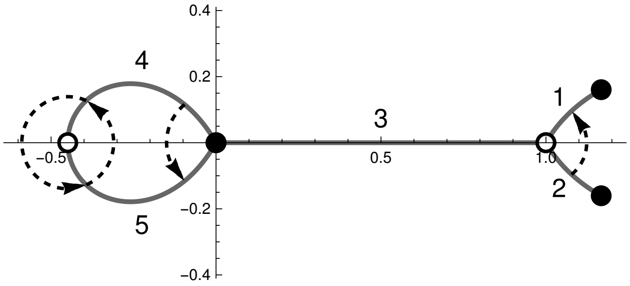


Figure 4.5.: The extending pattern of a Belyĭ map

The construction will be illustrated with an example. For the black vertices, the path 4 → 5 is contained in $\mathcal{R}_{-1/2}$ and crosses $\mathbb{P}^1(\mathbb{R})$ from upper half-plane to lower half-plane, so that $f_0(4) = a$. For all other edges i, $f_0(i) = 1$ since the other paths around black vertices are contained in $\mathcal{R}_{1/2}$. In the case of the white vertices, 4 → 5 is contained in $\mathcal{R}_{-1/2}$, crossing from upper half-plane to lower half-plane, while 5 → 4 crosses from lower half-plane to upper half-plane. Further, 2 → 1 lies in $\mathcal{R}_{3/2}$ and crosses from lower half-plane to upper half-plane. Thus,

$$f_1(i) = \begin{cases} 1 & \text{if } i = 1 \text{ or } 3, \\ b & \text{if } i = 2, \\ a & \text{if } i = 4, \\ a^{-1} & \text{if } i = 5. \end{cases}$$

4. Computing the Monodromy of a Composition

4.1. Computing the Extending Pattern

The approach taken to compute the extending pattern of β is largely the same as for computing its monodromy. However, instead of finding the label of the endpoint of a path, one must determine whether, and where, the path crosses $\mathbb{P}^1(\mathbb{R})$. Determining crossings is achieved by creating an interpolating function from the points on the path returned by the differential equation solver. This function is then sampled at sufficiently many points to find the location of any sign changes in the imaginary part of the function. Depending on the location and direction of the sign change of a path originating at an edge e_β, an element of F_2 is assigned to e_β. This process is performed for the liftings of both g_0 and g_1.

Algorithm 4.13 Computing the extending pattern of β

```
function ExtendingPattern($\beta$, num_samples)
$d \leftarrow \deg\beta$
preimages $\leftarrow$ solve($\beta - 0.5$, $x$)
rhs(a) $\leftarrow 2\pi i\frac{\beta-a}{\beta'}$
ext_pat $\leftarrow$ [[], []]
for $0 \le j \le d-1$ do
    $z_j \leftarrow$ preimages[j]
    for $0 \le a \le 1$ do
        $\tilde{g} \leftarrow$ solve($g' =$ rhs(a)$(g)$, $g(0) = z_j$, $0 \le t \le 1$)
        append PathAction($\tilde{g}$, num_samples) to ext_pat[$a$]
return ext_pat
```

4.2. Computing the Monodromy of the Composition

By viewing $\operatorname{Mon}\beta\gamma$ as a subgroup of $\operatorname{Mon}\gamma \wr_{E_\beta} \operatorname{Mon}\beta$ and making use of the extending pattern, computing the constellation of a composition of Belyǐ maps becomes rather simple.

If $\deg(\Delta_\beta) = n$ and $\deg(\Delta_\gamma) = m$, the constellation of $\beta\gamma$ consists of permutations on the mn edges $E_{\beta\gamma} \leftrightarrow E_\gamma \times E_\beta$. The identification $\{t\}_{t=0}^{mn-1} \leftrightarrow \{r\}_{r=0}^{m-1} \times \{s\}_{s=0}^{n-1}$ is given by $(r, s) \mapsto rn + s$. In the other direction, $k \mapsto (\lfloor k/n \rfloor, k \bmod n)$. From (4.3), g_j acts on the edges of $\Delta_{\beta\gamma}$ as

$$(r \cdot n + s)^{g_j} = (r, s)^{g_j} = \Big(r \cdot f_{g_j}(s), s\tau_{g_j}\Big) = n\Big(r \cdot f_{g_j}(s)\Big) + s\tau_{g_j}.$$

Because

```
for $0 \le r < m$ do
    for $0 \le s < n$ do
        $r * n + s$
```

generates the numbers $\{t\}_{t=0}^{mn-1}$ in sequential order,the permutations of $\Delta_{\beta\gamma}$ can be recovered by simply appending the image of $rn + s$ to a list.

Algorithm 4.14 Determining the action of a path

```
function PathAction($\tilde{g}$, num_samples)
action $\leftarrow 1$
samples $\leftarrow \left\{ \left( t, \tilde{g}(t) \right) \mid 1 \leq s \leq$ num_samples$, t = \frac{s-1}{\texttt{num\_samples}} \right\}$
for $0 \leq s <$ num_samples do
    if imag(samples[s]) * imag(samples[s+1]) $\leq 0$ then
        if imag(samples[s])$\geq 0$ and imag(samples[s+1])$< 0$ then
            if real(samples[s])$< 0$ then
                action *= $a$
            else if real(samples[s])$> 1$ then
                action *= $b^{-1}$
        else if imag(samples[s])$< 0$ and imag(samples[s+1])$\geq 0$ then
            if real(samples[s])$< 0$ then
                action *= $a^{-1}$
            else if real(samples[s])$> 1$ then
                action *= $b$
return action
```

5. The Group Structure of $\operatorname{Mon}\beta\gamma$

5.1. The Kernel of $\operatorname{Mon}\beta\gamma \to \operatorname{Mon}\beta$

While the last section allows one to determine generators of $\operatorname{Mon}\beta\gamma$ as a permutation group, one would often like to know more about the structure of $\operatorname{Mon}\beta\gamma$. Letting $A_\gamma := \ker(\operatorname{Mon}\beta\gamma \to \operatorname{Mon}\beta)$, the short exact sequence

$$1 \to A_\gamma \to \operatorname{Mon}\beta\gamma \to \operatorname{Mon}\beta \to 1,$$

provides a better description of the structure of $\operatorname{Mon}\beta\gamma$, so that the goal becomes determination of A_γ. More interesting than A_γ is

$$A := \ker\left(\varphi(\pi_1^Z) \to \operatorname{Mon}\beta\right),$$

as this provides A_γ for all γ simultaneously via the homomorphisms $\rho_{\gamma *}$.

Algorithm 4.15 Obtaining $\Delta_{\beta\circ\gamma}$ from Δ_β and Δ_γ

```
function ComposeConstellations($\{\tau_0, \tau_1, f_0, f_1\}$, $\{\sigma_0, \sigma_1\}$)
$\rho_{\gamma*} \leftarrow (a \mapsto \sigma_0, b \mapsto \sigma_1)$
$f_0, f_1 \leftarrow \rho_{\gamma*}(f_0), \rho_{\gamma*}(f_1)$
$n \leftarrow \deg \Delta_\beta$, $m \leftarrow \deg \Delta_\gamma)$
$\eta_0 \leftarrow [\,]$, $\eta_1 \leftarrow [\,]$
for $0 \le r < m$ do
    for $0 \le s < n$ do
        append $r^{f_0(s)} \cdot n + \tau_0(s)$ to $\eta_0$
        append $r^{f_1(s)} \cdot n + \tau_1(s)$ to $\eta_1$
return $\eta_0, \eta_1$
```

Lemma 4.16. *Let* $\mathrm{proj}_\beta : \mathrm{Fun}(E_\beta, \pi_1^Y) \rtimes \mathrm{Mon}\,\beta \to \mathrm{Mon}\,\beta$*. Then*

$$\varphi(\pi_1^Z) \approx \ker \mathrm{proj}_\beta \rtimes \mathrm{Mon}\,\beta.$$

Proof. Identify $\mathrm{Mon}\,\beta$ with $\{(1, \tau) \in \varphi(\pi_1^Z) \mid \tau \in \mathrm{Mon}\,\beta\}$. As $\ker \mathrm{proj}_\beta$ is normal and $\ker \mathrm{proj}_\beta \cap \mathrm{Mon}\,\beta = 1$, the result follows by the recognition theorem for semidirect products because given $(f, \tau) \in \varphi(\pi_1^Z)$, $(f, \tau) = (f, 1)(1, \tau)$. □

Writing G^n for $\mathrm{Fun}(E_\beta, G)$, where $n = |E_\beta|$, and A for $\ker \mathrm{proj}_\beta$, results in the following commutative diagram

$$\begin{tikzcd}
1 \arrow[r] & A \arrow[r] \arrow[d] & A \rtimes \mathrm{Mon}\,\beta \arrow[r, "\mathrm{proj}_\beta"] \arrow[d] & \mathrm{Mon}\,\beta \arrow[r] \arrow[d] & 1 \\
1 \arrow[r] & (\pi_1^Y)^n \arrow[r] & (\pi_1^Y)^n \rtimes \mathrm{Mon}\,\beta \arrow[r, "\mathrm{proj}_\beta"] & \mathrm{Mon}\,\beta \arrow[r] & 1
\end{tikzcd}$$

which is mapped onto

$$\begin{tikzcd}
1 \arrow[r] & \rho_{\gamma*}(A) \arrow[r] \arrow[d] & \mathrm{Mon}\,\beta\gamma \arrow[r, "\mathrm{proj}_\beta"] \arrow[d] & \mathrm{Mon}\,\beta \arrow[r] \arrow[d] & 1 \\
1 \arrow[r] & (\mathrm{Mon}\,\gamma)^n \arrow[r] & (\mathrm{Mon}\,\gamma)^n \rtimes \mathrm{Mon}\,\beta \arrow[r, "\mathrm{proj}_\beta"] & \mathrm{Mon}\,\beta \arrow[r] & 1
\end{tikzcd}$$

by $\rho_{\gamma*} \rtimes \mathrm{id}$. Thus, to understand $\mathrm{Mon}\,\beta\gamma$ for all γ, it suffices to determine $\ker \mathrm{proj}_\beta$.

5.2. Computing Generators of the Kernel

Lemma 4.17. *With notation as in Theorem 4.12 and Lemma 4.16,*

$$\ker \operatorname{proj}_\beta = \varphi(\ker \rho_\beta) \approx \ker \rho_\beta / \ker \varphi.$$

Proof. Recall from Theorem 4.12 that $\rho_\beta(\lambda) = \operatorname{proj}_\beta \varphi(\lambda)$:

$$\begin{tikzcd} F_2 \arrow[r, "\varphi"] \arrow[dr, "\rho_\beta"'] & A \rtimes \operatorname{Mon}\beta \arrow[d, "\operatorname{proj}_\beta"] \\ & \operatorname{Mon}\beta \end{tikzcd} .$$

As

$$\lambda \in \ker \rho_\beta \implies \lambda \in \ker \operatorname{proj}_\beta \varphi \implies \varphi(\lambda) \in \ker \operatorname{proj}_\beta,$$

$\varphi(\ker \rho_\beta) \leq \ker \operatorname{proj}_\beta$. But if $g \in \ker \operatorname{proj}_\beta$, then by the surjectivity of φ, there exists $\lambda \in F_2$ with $\varphi(\lambda) = g$. Moreover, $\rho_\beta(\lambda) = \operatorname{proj}_\beta \varphi(\lambda) = 1$ and $g \in \varphi(\ker \rho_\beta)$. Thus, noting that $\ker \varphi \leq \ker \rho_\beta$,

$$\ker \operatorname{proj}_\beta = \varphi(\ker \rho_\beta) \approx \ker \rho_\beta / \ker \varphi \tag{4.4}$$

by the first isomorphism theorem. □

Putting everything together yields

Theorem 4.18. *Let β be a dynamical Belyĭ map with constellation (τ_0, τ_1) and extending pattern (f_0, f_1). Let*

$$\varphi : \begin{array}{l} g_0 \mapsto (f_0, \tau_0) \\ g_1 \mapsto (f_1, \tau_1) \end{array}$$

and $A := \varphi(\ker \rho_\beta)$. Then for any Belyĭ map γ,

$$\operatorname{Mon} \beta\gamma \approx \rho_{\gamma*}(A) \rtimes \operatorname{Mon}\beta.$$

Although there are not specialized algorithms for computing $\ker \operatorname{proj}_\beta$, $\ker \rho_\beta$ can be efficiently computed, being a homomorphism from a free group to a permutation

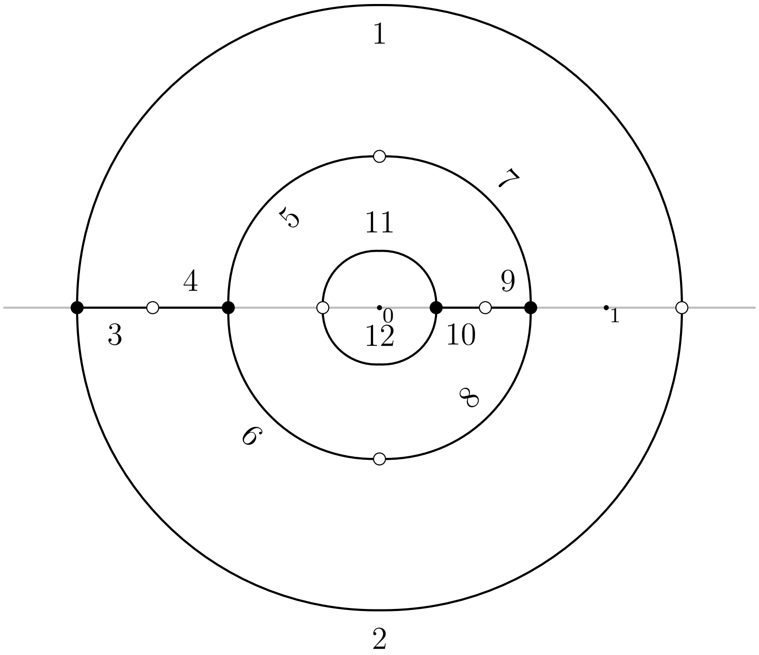


Figure 4.6.: A plot of β with edges labeled

group. To this end, computing the generators of $\ker \rho_\beta$ and finding their images under φ gives a set of generators for $\ker \operatorname{proj}_\beta$, though this set may not be minimal.

5.3. Example

Let

$$\beta_0(z) := \frac{(z^4 + 228z^3 + 494z^2 - 228z + 1)^3}{1728z(z^2 - 11z - 1)^5},$$
$$\mu(z) := \frac{55\sqrt{5} + 123}{5\sqrt{5} + 11} z = (11 + \alpha^{-1})z,$$

where α is the positive root of $z^2 - 11z - 1$. Let $\beta(z) = \beta_0\mu(z)$, and consider the family of Belyĭ maps $\gamma_m(z) = z^m$. As β_0 is not a dynamical Belyĭ map, μ moves the pole lying in the face containing $z = 1$ to lie at $z = 1$. The constellation and extending pattern of β (under some ordering of the edges) is given by

$$\tau_0 = (1\ 2\ 3)(4\ 6\ 5)(7\ 9\ 8)(10\ 11\ 12), \qquad f_0 = [a, a^{-1}, 1, a, 1, a^{-1}, 1, 1, 1, 1, 1, 1],$$
$$\tau_1 = (1\ 2)(3\ 4)(5\ 7)(6\ 8)(9\ 10)(11\ 12), \qquad f_1 = [b^{-1}, b, 1, 1, 1, 1, 1, 1, 1, 1, a, a^{-1}].$$

The constellation of γ is $\sigma_0 = (1\ \cdots\ n), \sigma_1 = 1$. As a result, applying $(a \mapsto a, b \mapsto 1) \rtimes \mathrm{id}$ before $\rho_{\gamma *} \rtimes \mathrm{id}$ does not change the image.

$$\Big(\rho_{\gamma *} \rtimes \mathrm{id}\Big) \circ \varphi = \Big(\rho_{\gamma *} \rtimes \mathrm{id}\Big) \circ \Big((a \mapsto a, b \mapsto 1) \rtimes \mathrm{id}\Big) \circ \varphi$$

That is, φ can be replaced by

$$\Big((a \mapsto a, b \mapsto 1) \rtimes \mathrm{id}\Big) \circ \varphi,$$

without affecting $\ker \Big(A_\gamma \rtimes \operatorname{Mon} \beta \to \operatorname{Mon} \beta\Big)$, resulting in $\varphi(\pi_1^Z) \leq F_1 \wr \operatorname{Mon} \beta$ and a simpler computation of $\ker \mathrm{proj}_\beta$[1].

To begin, form F_2, form $\operatorname{Mon} \beta$, and find the homomorphism between them.

```
sage:  F2 = libgap.FreeGroup('a', 'b'); A,B = F2.GeneratorsOfGroup()

sage:  tau0 = libgap.eval('(1,2,3)(4,6,5)(7,9,8)(10,11,12)')

sage:  tau1 = libgap.eval('(1,2)(3,4)(5,7)(6,8)(9,10)(11,12)')

sage:  MonBeta = libgap.Group(tau0, tau1)

sage:  rho = F2.GroupHomomorphismByImages(MonBeta)
```

Then form the wreath product, embed the extending pattern in $\mathrm{Fun}(E_\beta, F_2) \rtimes 1$, and form $\operatorname{Mon} \beta\gamma$.

[1] In the case that $\varphi(\pi_1^Z)$ can be viewed as a subgroup of $F_1 \wr \operatorname{Mon} \beta$, either if $\gamma(z) = z^n$ as here or if the edges of β cross only one of $(-\infty, 0)$ or $(1, \infty)$, using the Residue-Class-Wise Affine Groups [35] package for GAP provides a superior method to the approach presented.

```
gap> LoadPackage("rcwa");;
gap> tau0:=(1,2,3)(4,6,5)(7,9,8)(10,11,12);;
gap> tau1:=(1,2)(3,4)(5,7)(6,8)(9,10)(11,12);;
gap> MonBeta:=Group(tau0, tau1);;
gap> F1:=CyclicGroup(IsRcwaGroupOverZ,infinity);
gap> wr:=WreathProduct(F1, MonBeta);;
gap> gens:=WreathProductInfo(wr).basegens;;
gap> f0:=gens[1] * gens[2]^-1 * gens[4] * gens[6]^-1;;
gap> f1:=gens[11] * gens[12]^-1;;
gap> MonBetaGamma:=Group(f0*wr.3, f1*wr.4);;
gap> K:=KernelOfActionOnRespectedPartition(MonBetaGamma);;
gap> List(GeneratorsOfGroup(K), Factorization);
[ [ ClassShift( 10(12) )^-5, ClassShift( 0(12) )^5 ],
  [ ClassShift( 11(12) )^-5, ClassShift( 1(12) )^5 ],
  [ ClassShift( 9(12) )^-5, ClassShift( 2(12) )^5 ],
  [ ClassShift( 8(12) )^-5, ClassShift( 3(12) )^5 ],
  [ ClassShift( 6(12) )^-5, ClassShift( 4(12) )^5 ],
  [ ClassShift( 7(12) )^-5, ClassShift( 5(12) )^5 ] ]
```

```
sage:  wr = F2.WreathProduct(MonBeta)
sage:  a_gens = [A^wr.Embedding(j) for j in range(1, 13)]
sage:  f0 = a_gens[0] * a_gens[1]^-1 * a_gens[3] * a_gens[5]^-1
sage:  f1 = a_gens[10] * a_gens[11]^-1
sage:  tau0, tau1 = wr.Embedding(13).Image().GeneratorsOfGroup()
sage:  MonBetaGamma = libgap.Group(f0*tau0, f1*tau1)
```

Find the image of φ restricted to $\ker \rho$; this will be $\ker \mathrm{proj}_{\beta}$.

```
sage:  phi = F2.GroupHomomorphismByImages(MonBetaGamma)
sage:  KerProjBeta = phi.RestrictedMapping(rho.Kernel()).Image()
```

Due to a lack of efficient methods for finding generators in a wreath product, GAP is unable to find a small generating set from the 61 generators of KerProjBeta, or even from the 15 unique generators of KerProjBeta. As a result, this must be done manually. In this case, all 15 of the unique generators can be expressed as

$$\prod_{i=1}^{6} g_i^{e_i}, \qquad e_i \in \{0, 1, -1\}$$

where $\{g_i\}_{i=1}^{6}$ is a fixed set of six of the generators.

```
sage:  gens = list(KerPBeta.GeneratorsOfGroup().Unique())
sage:  exponents_0_pm_1 = ( (wr.One(), gen, gen^-1) for gen in gens[1:7] )
sage:  gen_and_inv_prods = map(prod, itertools.product(*minimal_gens))
sage:  set(gens).issubset(gen_and_inv_prods)
True
sage:  gens[1:7][2]
[WreathProductElement(<id>, a^5, <id>, <id>, <id>, <id>,
                      <id>, <id>, <id>, <id>, <id>, a^-5, ()),
 WreathProductElement(a^-5, <id>, <id>, <id>, <id>, <id>,
                      <id>, <id>, <id>, <id>, a^5, <id>, ()),
 WreathProductElement(<id>, <id>, a^-5, <id>, <id>, <id>,
```

[2] The output of this command was formatted to improve readability.

```
                     <id>, <id>, <id>, a^5, <id>, <id>, ()),
WreathProductElement(<id>, <id>, <id>, a^-5, <id>, <id>,
                     <id>, <id>, a^5, <id>, <id>, <id>, ()),
WreathProductElement(<id>, <id>, <id>, <id>, a^-5, <id>,
                     a^5, <id>, <id>, <id>, <id>, <id>, ()),
WreathProductElement(<id>, <id>, <id>, <id>, <id>, a^-5,
                     <id>, a^5, <id>, <id>, <id>, <id>, ())]
```

As such $\ker \operatorname{proj}_\beta \approx F_1^6 \leq (\pi_1^Y)^{12}$ and for any $\gamma(z) = z^n$,

$$\begin{gathered} F_1^6 \rtimes A_5 \xrightarrow{\sim} \operatorname{Mon}\beta\gamma \leq \operatorname{Mon}\gamma \wr_{E_\beta} \operatorname{Mon}\beta \\ (x^{5a_1}, \ldots, x^{5a_6}, \tau) \mapsto (\rho_\gamma(x)^{5a_1}, \ldots, \rho_\gamma(x)^{5a_6}, \tau) = \Big((1 \cdots n)^{5a_1}, \ldots, (1 \cdots n)^{5a_6}, \tau\Big) \end{gathered}$$

In particular, for $\gamma(z) = z^n$,

$$\operatorname{Mon}\beta\gamma \approx \begin{cases} C_n^6 \rtimes A_5 & \text{if } 5 \nmid n, \\ C_{n/5}^6 \rtimes A_5 & \text{if } 5 \mid n. \end{cases}.$$

CHAPTER 5. CRYPTOGRAPHY

While computing a Belyĭ map from a given constellation is difficult, computing the constellation from the Belyĭ map is much easier. As such, it is natural to attempt to use dessins d'enfants for cryptography.

1. A First Approach

In the last chapter, it was shown how to efficiently compute $\Delta_{\beta\gamma}$ from Δ_γ, Δ_β, and the extending pattern of β. Suppose one party, Alice, knows a dynamical Belyĭ map β, while another party, Bob, knows a Belyĭ map γ. If Alice tells Bob Δ_β and the extending pattern of β, and Bob tells Alice Δ_γ, then both Alice and Bob can compute $\Delta_{\beta\gamma}$. However, for sufficiently complicated Belyĭ maps β, γ, it would be infeasible to recover either Belyĭ map from its constellation. Thus, both Alice and Bob know $\Delta_{\beta\gamma}$, though neither Alice nor Bob have revealed their private Belyĭ map.

2. Developing a Protocol

Composition of Belyĭ maps will now be used to construct a public-key authentication protocol. Suppose Δ_α is a constellation which Alice has made public and for which Alice knows a Belyĭ map α. Let Ali be a party which may or may not be Alice, and let Ali iniate a conversation with Bob. As it is computationally infeasible to compute a sufficiently complicated Belyĭ map from its constellation, if Ali can prove knowledge of α, then Bob should believe that Ali is indeed Alice. In order to prove knowledge of a Belyĭ map α without revealing it, Bob will send Ali a dynamical Belyĭ map β and will ask Ali to compose it with α. If, upon being given an arbitrary dynamical Belyĭ map β, Ali can respond with $\beta\alpha$, then one should believe that Ali knows α.

$$\begin{tikzcd}
A \arrow[r, shift right, "\beta\alpha"'] & B \arrow[l, shift right, "\beta"']
\end{tikzcd}$$

The difficulty comes in verifying that the response f to Bob's challenge is indeed the composition of β with a Belyĭ map having Δ_α as its constellation. In particular, `Monodromy` does not suffice to prove this because it works numerically and it is easy to find a numerical approximation to a Belyĭ map. Thus, a malicious third party, Mallory, would need only compute a numerical approximation α_m to α in order that `Monodromy` would find $\Delta_{\beta\alpha}$ as the constellation of $\beta\alpha_m$.

Moreover, Alice's Belyĭ map cannot be of genus zero as there are efficient algorithms for decomposing compositions of single-variable rational functions [1, 3]. Although there are also efficient algorithms for decomposing compositions of a single-variable rational function and a multivariate rational function [24], these algorithms are designed for rational functions over $\mathbb{P}^1(\mathbb{C})$, not curves of positive genus. While there are approaches using Gröbner bases for decomposing rational functions defined on an algebraic curve, "the complexity increases dramatically" [25] when working over an algebraic curve.

3. Verifying Constellations

Because `Monodromy` is insufficient to verify knowledge of a Belyĭ map with a given constellation, an additional step is necessary. The goal of the verification process is to ascertain whether Ali has knowledge of an exact representation for the Belyĭ map $\beta\alpha$. This additional step must, therefore, make use of the field over which the Belyĭ map is defined, forcing the use of symbolic methods. Conclusively identifying the monodromy group of a Belyĭ map through symbolic methods is difficult. Let f be the response by Ali to Bob's challenge β. Instead of determining conclusively that $\Delta_f \sim \Delta_{\beta\alpha}$, the additional step will instead determine whether there is a sufficiently large probability that $\operatorname{Mon} f$ is neither S_n nor A_n. Given the scarcity of Belyĭ maps with monodromy group other than S_n or A_n, along with the fact that $\operatorname{Mon} \beta\alpha \neq S_n, A_n$ by Lemma 5.5, if $\operatorname{Mon} f \neq S_n, A_n$ and `Monodromy` returns $\Delta_{\beta\alpha}$, it is probable that f indeed has constellation $\Delta_{\beta\alpha}$.

The approach proceeds as follows. Let β be a Belyĭ map on a hyperelliptic curve X defined by $f(x,y)=0$. Then, as will be seen,

$$\operatorname{Mon}\beta = \operatorname{Gal}\Big(\beta - t \in \mathbb{C}(t)(X)\Big) \trianglelefteq \operatorname{Gal}\Big(\beta - t \in K(t)(X)\Big).$$

For almost all $a \in K$, specializing the fields of the latter extension results in an isomorphic Galois group $\operatorname{Gal}\Big(\beta - a \in K(X)\Big)$. The equations $\beta(x,y)=a$ and $f(x,y)=0$ will be used to obtain a polynomial $g(x)$ so that $\operatorname{Gal}\Big(\beta - a \in K(X)\Big) \approx \operatorname{Gal}\Big(g(x)\Big)$. Then, factoring g modulo several primes $\mathfrak{p} \subseteq K$ gives the cycle types of elements of $\operatorname{Gal}\Big(g(x)\Big)$ and can be used to determine if $\operatorname{Gal}\Big(\beta - a \in K(X)\Big) \in \{S_n, A_n\}$. By normality, this also determines whether $\operatorname{Mon}\gamma \in \{S_n, A_n\}$.

3.1. Identifying S_n and A_n as a Monodromy Group

3.1.1. Algebraic Monodromy Group

Definition. *Let $x := (x_1, \ldots, x_n)$ and let $p(x)/q(x)$ be a rational function $X \to \mathbb{P}^1(\mathbb{C})$ with field of definition K. Let L be the splitting field of $p - t\,q$ over $K(t)$. The algebraic monodromy group of p/q is $\operatorname{AMon} p/q := \operatorname{Gal}\Big(L/K(t)\Big)$ and the geometric monodromy group is $\operatorname{GMon} p/q := \operatorname{Gal}\Big(LK^{\mathrm{a}}/K(t)\Big)$.*

First are presented some theorems to relate monodromy, geometric monodromy, and algebraic monodromy in the case of a Belyĭ map.

Theorem 5.1. *[26] Let $h : X \to Y$ be a covering map of complex algebraic varieties of degree n and let L be the Galois closure of $\mathbb{C}(X)$ over $\beta^*\mathbb{C}(Y)$, where $\beta^* f = f\beta$. Then $\operatorname{Mon}\beta = \operatorname{Gal}\Big(L/\beta^*\mathbb{C}(Y)\Big)$ as subgroups of S_n.*

Theorem 5.2. *[42] Let $K \subseteq \mathbb{C}$ be a subfield and let K^{a} be an algebraic closure of K in $\mathbb{C}$. Let $\beta : X \to Y$ be a finite unramified covering of a complex algebraic variety, with Y defined over K^{a}. If $\mathbb{C}(X)/\beta^*\mathbb{C}(Y)$ is Galois, then so is $K^{\mathrm{a}}(X)/\beta^* K^{\mathrm{a}}(Y)$ and their Galois groups are isomorphic.*

Let X be the hyperelliptic curve defined by $f(x,y)=0$ and let $\beta = P/Q$, where $P, Q \in K[X]$, be a Belyĭ map on X of degree n defined over K. Let L be the splitting field of $P(x,y) - t\,Q(x,y) \in K(t)[X]$ and let Z satisfy $\mathbb{C}(Z) \approx L$. Then by Theorem 5.1, $\operatorname{Mon}\beta \approx \operatorname{Gal}\left(\mathbb{C}(Z)/\beta^*\mathbb{C}(t)\right)$, and by Theorem 5.2, $\operatorname{Mon}\beta \approx \operatorname{Gal}\left(K^{\mathrm{a}}(Z)/K^{\mathrm{a}}(t)\right)$, noting that $\mathbb{P}^1(\mathbb{C})$ is defined over K^{a}. That is,

Corollary. *For a rational covering map $h : X \to \mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$,*

$$\operatorname{Mon} h \approx \operatorname{GMon} h.$$

Moreover, $\operatorname{GMon}\beta \trianglelefteq \operatorname{AMon}\beta$, hence $\operatorname{Mon}\beta \trianglelefteq \operatorname{AMon}\beta$.

Theorem 5.3. *[62, Prop. 1.13] For a rational covering map $h : X \to \mathbb{P}^1(\mathbb{C})\backslash\{0,1,\infty\}$,*

$$\operatorname{Mon} h \trianglelefteq \operatorname{AMon} h.$$

Let $\hat{K} := L \cap K^{\mathrm{a}}$. The theorem follows from the following diagram which relates $\operatorname{GMon} h$ to $\operatorname{Gal}\left(L/\hat{K}(t)\right)$.

$$\begin{tikzcd}
 & & LK^{\mathrm{a}} \arrow[dl, dash, "H"] \arrow[dr, dash] & \\
 & K^{\mathrm{a}}(t) \arrow[dl, dash] \arrow[dr, dash] & & L \arrow[dl, dash, "H"'] \arrow[dd, dash, "G"] \\
K^{\mathrm{a}} \arrow[dr, dash] & & \hat{K}(t) \arrow[dl, dash] \arrow[dr, dash] & \\
 & \hat{K} \arrow[dr, dash] & & K(t) \arrow[dl, dash] \\
 & & K &
\end{tikzcd}$$

For an alternative account of this discussion, see also [30, pg. 58-59].

3.1.2. Implications of $\operatorname{AMon} f \in \{S_n, A_n\}$

In this section, $n := \deg f$ and it is assumed that $\operatorname{Mon} f \neq 1$. For $n \geq 5$, A_n is simple [19, Cor. 3.3A] and the only normal subgroups of S_n are $1, A_n, S_n$.

Lemma 5.4. $\operatorname{Mon} f \in \{S_n, A_n\}$ *iff* $\operatorname{AMon} f \in \{S_n, A_n\}$.

Proof. By Theorem 5.3, if $\operatorname{AMon} f \in \{S_n, A_n\}$, then because $\operatorname{Mon} f \trianglelefteq \operatorname{AMon} f$, $\operatorname{Mon} f \in \{S_n, A_n\}$. On the other hand, if $\operatorname{Mon} f \in \{S_n, A_n\}$, then because $\operatorname{Mon} f \leq \operatorname{AMon} f$, also $\operatorname{AMon} f \in \{S_n, A_n\}$. □

Therefore, it suffices to determine that $\operatorname{AMon} f \in \{S_n, A_n\}$ in order to conclude that $\operatorname{Mon} f \in \{S_n, A_n\}$. But $\operatorname{Mon} \beta\alpha \notin \{S_n, A_n\}$ as the next lemma shows, so that if $\operatorname{Mon} f \in \{S_n, A_n\}$, $f \neq \beta\alpha$.

Lemma 5.5. *If $m, n > 1$, $|S_m \wr S_n| < \frac{(nm)!}{2}$.*

Proof. Note that $|S_m \wr S_n| = (m!)^n n!$. The proof proceeds by double induction on m and n. For $m, n = 2$,

$$(m!)^n n! = (2)^2 \cdot 2 = 8 < 12 = \frac{4!}{2} = \frac{(mn)!}{2}.$$

Now, suppose that $\frac{(n_0 m_0)!}{2} > (m_0!)^{n_0} n_0!$ for some m_0, n_0. Then

$$\frac{[(m_0+1)!]^{n_0} n_0!}{[m_0!]^{n_0} n_0!} = [(m_0+1)]^{n_0}, \qquad \frac{[(m_0+1)n_0!]}{(m_0 n_0)!} = (m_0 n_0 + n_0) \cdots (m_0 n_0 + 1)$$

shows that $\frac{(n_0 m)!}{2} > (m!)^{n_0} n_0!$ for all $m > m_0$. Thus, for all $m > m_0$,

$$\frac{(m!)^{n_0+1}(n_0+1)!}{(m!)^{n_0} n_0!} = (n_0+1)m! = (mn_0+m)(m-1)!,$$
$$\frac{[m(n_0+1)]!}{(mn_0)!} = (mn_0+m)(mn_0+m-1)\cdots(mn_0+1),$$

so that for all $n > n_0$, $\frac{(nm)!}{2} > (m!)^n n!$. □

3.1.3. Specialization

In order to apply the results of the last subsection, it is necessary to be able to determine if $\operatorname{AMon} f \in \{S_n, A_n\}$. Attempting to determine this from the definition is difficult. The first step will be to eliminate a variable using resultants.

Definition. *[69] Let $f, g \in R[x]$, where R is a ring. Then the resultant in x res_x of f and g is the smallest polynomial in the coefficients of f and g which vanishes iff f and g have a common zero.*

The resultant of $f := \sum_{i=0}^{m} a_i x^i$ and $g := \sum_{i=0}^{n} b_i x^i$ can be calculated as the determinant of the matrix

$$\begin{bmatrix} a_m & & \cdots & & a_0 & & \\ & \ddots & & & & \ddots & \\ & & a_m & & \cdots & & a_0 \\ b_n & & \cdots & & b_0 & & \\ & \ddots & & & & \ddots & \\ & & b_n & & \cdots & & b_0 \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} n \text{ rows} \\ \left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} m \text{ rows} \end{matrix} .$$

The result of taking the determinant will be a homogeneous polynomial of degree $m+n$ in the coefficients of f and g. In particular, $\operatorname{res}_x(f,g)$ is independent of x.

Let $f(x,y)$ be an equation defining a hyperelliptic curve with vanishing set X and let β be a Belyĭ map with domain X. If $\deg \beta = n$, for any $t \notin \{0,1,\infty\}$, $\beta = t$ will have n roots. Because X is a hyperelliptic curve, β can be put in the form

$$\beta(x,y) := \frac{p(x) + y\,q(x)}{r(x)},$$

so that for $q(x) \neq 0$, $\beta(x,y) = t$ implies that

$$y = \frac{t\,r(x) - p(x)}{q(x)} \tag{5.1}$$

and the roots of $\beta = t$ on X are solutions to

$$0 = q(x)^2 f\Big(x, \frac{t\,r(x) - p(x)}{q(x)}\Big) \in \mathbb{C}(t)[x].$$

In fact, from (5.1), finding the roots of this latter equation determines the solutions to $\beta(x,y) = t$.

It turns out that this last expression is exactly $\operatorname{res}_y\Big(f(x,y), y\,q(x) + p(x) - t\,r(x)\Big)$

$$\begin{vmatrix} 1 & f_1(x) & f_0(x) \\ q & p - r\,t & 0 \\ 0 & q & p - r\,t \end{vmatrix},$$

where $f(x,y) = y^2 + f_1(x)\,y + f_0(x)$ with $2\deg f_1 < \deg f_0$. However, any roots of $r(x)$ can lead to extra factors in the resultant corresponding to roots which $r(x)$ has in common with $y\,q(x)+p(x)$. That is, instead of $g_0(x,t) := \operatorname{res}_y\Big(f(x,y), y\,q(x) + p(x) - t\,r(x)\Big)$,

$$\beta = t \Longleftrightarrow g(x,t) := \frac{g_0(x,t)}{\gcd\Big(g_0(x,t), \operatorname{res}_y\Big(y\,q(x)+p(x), r(x)\Big)\Big)} = 0.$$

Denote by $\operatorname{Gal}(\beta - t)$ the Galois group of the splitting field of $\beta(x,y) - t$ as an algebraic extension of the quotient field L of $K(t)[X]$. Because the right-hand side of (5.1) lies in L, the roots of $\beta(x,y)-t$ are in L-rational bijection with the roots of $g(x,t) = 0$. Hence

$$\operatorname{Gal}(\beta - t) \approx \operatorname{Gal}\Big(g(x,t)\Big).$$

Finally, Hilbert's irreducibility thorem allows for specializing $g(x,t)$ while ensuring that the resulting Galois group is isomorphic to $\operatorname{Gal}\Big(g(x,t) \in K(t)[x]\Big)$. Upon specializing, one is left with a univariate polynomial $g(x)$ over a number field, for which it will be shown how to determine whether $\operatorname{Gal}\Big(g(x)\Big) \in \{S_n, A_n\}$. Being a single-variable polynomial, $g(x)$ can be efficiently factored modulo a prime ideal $\mathfrak{p} \subseteq K$ [50].

A method for choosing specializations t_0 of t which will result in an isomorphic Galois group [56, Sec. 4.6]. A general framework for choosing specializations can be found in [37].

3.2. Identifying S_n and A_n as a Galois Group

Definition. *[39, pg. 341] Let A be integrally closed in its quotient field K, and let B be its integral closure in a finite Galois extension L. Let $\mathfrak{P}$ be a maximal ideal of B. The decomposition group of $\mathfrak{P}$ is*

$$G_{\mathfrak{P}} := \{\sigma \in \operatorname{Gal}(L/K) \mid \mathfrak{P}^\sigma = \mathfrak{P}\}.$$

Let $\overline{B} := B/\mathfrak{P}$ and $\overline{A} := A/\mathfrak{p}$, where $\mathfrak{P} \cap A = \mathfrak{p}$. Note that L/K is separable since K has characteristic zero and L/K is finite. Thus, there is a homomorphism $G_{\mathfrak{P}} \to \operatorname{Gal}(\overline{B}/\overline{A})$ given by $\sigma \mapsto \overline{\sigma}$, where $\overline{\sigma}$ is defined by

$$\overline{x}^{\overline{\sigma}} = \overline{x^\sigma}.$$

Let the image of this homomorphism be denoted $\overline{G_{\mathfrak{P}}}$.

Theorem 5.6. *[39, Thm. 7.2.9] Let A be an integral domain, integrally closed in its quotient field k. Let $f(x) \in A[x]$ be monic and irreducible over k. Let $\mathfrak{p}$ be a maximal ideal of A, let $\overline{f} = f \bmod \mathfrak{p}[x]$. Suppose that $\overline{f}$ has no multiple roots in an algebraic closure of $A/\mathfrak{p}$. Let L be a splitting field for f over k, and let B be the integral closure of A in L. Let $\mathfrak{P}$ be any prime of B over $\mathfrak{p}$ and let a bar denote reduction* $\mathrm{mod}\,\mathfrak{p}$*. Then the map*

$$G_{\mathfrak{P}} \to \overline{G_{\mathfrak{P}}}$$

is an isomorphism of $G_{\mathfrak{P}}$ with the Galois group of $\overline{f}$ over $\overline{A}$.

Let $\mathfrak{p}$ and $\mathfrak{P}$ satisfy the hypotheses of Theorem 5.6. Note that $G_{\mathfrak{P}} \to \overline{G_{\mathfrak{P}}}$ induces an isomorphism of group actions as follows. Let

$$R := \text{roots of } f = \{x_1, \ldots, x_n\}, \qquad \overline{R} := \text{roots of } \overline{f} = \{\overline{x_1}, \ldots, \overline{x_n}\}.$$

Then $S_{\overline{R}} \approx S_R$ by $x_i^{\sigma} = \overline{x_i}^{\overline{\sigma}}$. Let $x \in A$, $\overline{x} := x \bmod \mathfrak{p}$, $\overline{\sigma} \in S_{\overline{R}}$, and σ be the preimage of $\overline{\sigma}$. That is,

$$\begin{array}{ccccc}
\overline{G_{\mathfrak{P}}} & \to & S_{\overline{R}} & \to & S_R \\
\overline{\sigma} & \mapsto & \left(\overline{x} \mapsto \overline{x}^{\overline{\sigma}}\right) & \mapsto & \left(x \mapsto x^{\sigma}\right),
\end{array}$$

$$\begin{array}{ccccc}
\overline{G_{\mathfrak{P}}} & \to & \mathrm{Gal}(L/K) & \to & S_R \\
\overline{\sigma} & \mapsto & \sigma & \mapsto & \left(x \mapsto x^{\sigma}\right)
\end{array}$$

and the following diagram commutes.

$$\begin{tikzcd}
\overline{G_{\mathfrak{P}}} \arrow[r] \arrow[d] & S_{\overline{R}} \arrow[d] \\
\mathrm{Gal}(L/K) \arrow[r] & S_R
\end{tikzcd}$$

In particular, $\bar{\ }$ gives an isomorphism of group actions and given any permutation in $\mathrm{im}\left(\overline{G_{\mathfrak{P}}} \to S_{\overline{R}}\right)$, there is a permutation in $\mathrm{im}(\mathrm{Gal}(L/K) \to S_R)$ with the same cycle type.

That is, in order to determine cycle types of elements in $\mathrm{Gal}(L/K)$, it suffices to factor f modulo various primes $\mathfrak{p} \subseteq A$ (See [20, pg. 640-641]).

Having determined a collection of cycle types which appear in $\mathrm{Gal}(L/K)$ by specializing at several values of $t \neq 0, 1, \infty$ and factoring modulo several primes, one then wants to determine whether $\mathrm{Gal}(L/K)$ is S_n or A_n. There are several standard results which characterize S_n, A_n by the presence of a pair cycle types [13].

A more powerful characterization comes from looking at the r-transitivity of the group. By the classification of finite simple groups, the only groups which are r-transitive for $r > 3$ are S_n, A_n, and the Mathieu groups $M_{11}, M_{12}, M_{23}, M_{24}$ [19, pg. 218]. However, while S_n is n transitive and A_n is $n-2$-transitive, the Mathieu groups are not 6-transitive [19, pg. 34, 218]. As a result, to determine that $\mathrm{Gal}(L/K) \in \{S_n, A_n\}$, one need only show that $\mathrm{Gal}(L/K)$ is 6-transitive. The following approach to showing this is presented in [15]. Let G be a transitive permutation group of degree n. Given the existence of a prime p, $n/2 < p < n-2$, such that G has an element of order p, G is $n-p+1$-transitive. The obstruction to using $p \leq n/2$ is the possibility of the existence of a block of size $n/2$.

Consider, for example,

$$G := \langle (1\ 2\ 3), (4\ 5\ 6), (1\ 4)(2\ 5)(3\ 6) \rangle, \qquad K_0 := \langle (1\ 2\ 3) \rangle.$$

The proof of $n-p+1$-transitivity constructs subgroups which are v-transitive on their support of size $p+v$, using that for some $\sigma \in G$,

$$\mathrm{supp}(K_v) \cap \mathrm{supp}(K_v)^{\sigma} \notin \{\emptyset, \mathrm{supp}(K_v)\}.$$

Because $\mathrm{supp}(K_0)$ is a block of G, this fails for all σinG. If it can be guaranteed that $\mathrm{supp}(K_v)$ is not a block of G, then the proof can proceed.

Theorem 5.7. *[15] Let G be a transitive permutation group of degree n and $r|n$. If G contains an element x satisfying*

There is a c-cycle in the disjoint cycle
decomposition of x, where $(c, r!) = 1$ and $c > n/r$.

Then G does not have a block of size n/r.

To put this in context, assume that there exist primes $\mathfrak{p}_1, \mathfrak{p}_2$ so that $f \bmod \mathfrak{p}_1$ has an irreducible factor of odd length $c_1 > n/2$ and $f\mathfrak{p}_2$ has an irreducible factor of length c_2 with $(c_2, 6) = 1$ and $c_2 > n/3$. Then $\mathrm{Gal}(f)$ has no blocks of size $n/2$ or $n/3$, and the existence of a degree p, $p > n/4$, irreducible factor of $f \bmod \mathfrak{p}$ imples that $\mathrm{Gal}(f)$ is $n - p + 1$-transitive. If $n/4 < p \leq n - 5$, then $\mathrm{Gal}(f)$ is at least 6-transitive and is S_n or A_n.

There is a discussion of the expected frequency of occurence of permutations in S_n with no cycle of length r in its disjoint cycle decomposition. This can be used to determine the probability that factoring an irreducible polynomial modulo a given number of primes would fail to find that G 6-transitive if G is S_n or A_n.

Algorithm 5.8 Determining if the monodromy group is S_n or A_n

```
function IsMonodromy$S_nA_n$($\beta$, $f$, $\varepsilon$)
res1 ← Resultant(Numerator($\beta$) - $t$Denominator($\beta$), $f$, $y$)
res2 ← Resultant(Numerator($\beta$), Denominator($\beta$), $y$)
$g$ ← $\dfrac{\texttt{res1}}{\texttt{gcd(res1, res2)}}$
$g(x) \leftarrow g(x, t_0)$, where $t_0$ satisfies $\mathrm{Gal}\big(g(x,t)\big) \approx \mathrm{Gal}\big(g(x,t_0)\big)$
cycle_types ← []
while $\Pr(\mathrm{Mon}\,\beta \in \{S_n, A_n\}) \geq \varepsilon$ do
    choose a prime $\mathfrak{p}$ of $K$ so that $\mathrm{Gal}\big(g(x)\big) \approx \mathrm{Gal}\big(\overline{g(x)}\big)$
    factor $\overline{g(x)}$ and append list of factor degrees to cycle_types
    using Theorem 5.7, determine from cycle_types if $\mathrm{Gal}\big(\overline{g(x)}\big)$ is 6-transitive
    if $\mathrm{Gal}\big(\overline{g(x)}\big)$ is 6-transitive then
        return $\mathrm{Mon}\,\beta \in \{S_n, A_n\}$
return $\Pr(\mathrm{Mon}\,\beta \in \{S_n, A_n\}) < \varepsilon$
```

3.3. Occurence of S_n and A_n as Galois and Monodromy Groups

Van der Waerden proved in 1934 [63] that almost all monic irreducible polynomials $f(x) \in \mathbb{Z}[x]$ have $\mathrm{Gal}(f) = S_n$. Let

$$E_n(N) := \#\{f(x) \in \mathbb{Z}[x] \mid \deg f = n,\ f \text{ monic, height}(f) \leq N,\ \mathrm{Gal}(f) \neq S_n\}.$$

Taking into account also the reducible polynomial, in 1936, Van der Waerden gave the estimate [64]

$$E_n(N) \ll N^{n-c/\log\log N}, \quad c = \frac{n-2}{6}.$$

This was improved to

$$E_n(N) \ll N^{n-1/2}(\log N)^{1-\gamma_n}, \quad \gamma_n \sim (2\pi n)^{-1/2}$$

by Gallagher [21]. In the case that $\mathrm{Gal}(f) \neq S_n, A_n$, Dietmann proves that for $n \geq 9$ [18],

$$E_n(N)' \ll_{n,\varepsilon} N^{n-1+b(n)+\varepsilon}, \quad b(n) = 2\Big/\binom{n}{\lfloor n/2 \rfloor},$$

where

$$E_n(N)' := \#\{f(x) \in \mathbb{Z}[x] \mid \deg f = n,\ f \text{ monic, height}(f) \leq N,\ \mathrm{Gal}(f) \neq S_n, A_n\}.$$

Consider the case that $n = 20, N = 500$. There are 1001^{20} polynomials $f \in \mathbb{Z}[x]$ with $\deg f = 20$ and $|a_i| \leq 500$ for all coefficients a_i of f. Then by Dietmann's result, for all $\varepsilon > 0$, there is a constant $C_\varepsilon > 0$ so that

$$\Pr\Big(\mathrm{Gal}(f) \neq S_n, A_n\Big) \leq \frac{500^{19+b(20)}}{1001^{20}} \cdot 500^\varepsilon C_\varepsilon \approx 1.8697 \cdot 10^{-9} \cdot 500^\varepsilon C_\varepsilon.$$

Using Gallagher's result with $\gamma_n := 0$, there is a constant $C > 0$ so that

$$\Pr\Big(\mathrm{Gal}(f) \neq S_n\Big) \leq 2.5980 \cdot 10^{-7} \cdot C.$$

In addition to considering only the quantity, one may put a topology on $K[x]$ and investigate the occurence of S_n from a topological perspective. An interesting result in this direction is given by Heintz.

Theorem 5.9. *[27] Let K be a Hilbertian field. There exists a constant $c > 0$ with the following property: let $F \in K[X]$ be an arbitrary polynomial of degree $\deg F = d$, considered as a point of K^{d+1} with the Zariski topology. In each neighborhood of F, there exists a separable polynomial $G \in K[X]$ with $\deg G = d$, G has symmetric Galois group, and*

$$\begin{array}{c} \textit{number of} \\ \textit{multiplications} \\ \textit{required to evaluate} \\ G(X) \end{array} \leq 5 + 3 \cdot \begin{array}{c} \textit{number of} \\ \textit{multiplications} \\ \textit{required to evaluate} \\ F(X) \end{array}.$$

As an example of the disparity in the number of operations required to evaluate a polynomial, consider

$$f(x) = x^n - 1, \qquad g(x) = \sum_{j=1}^{n} 2^{2^j} x^j.$$

Let $L(h)$ denote the number of operations required to evaluate a polynomial h. Then [59]

$$L(f) \leq 2\lceil \log_2 n \rceil + 1, \qquad L(g) > \sqrt{n/3 \log_2 n}$$

For $n = 10^8$, $L(f) \leq 55$, while $L(g) > 1120$. Thus, not only do almost all polynomials have Galois group S_n, but in each complexity class for L, such polynomials are dense in the Zariski topology.

Further, the experimental evidence shown in Table 5.1 indicates that as with the Galois groups of $f(x) \in \mathbb{Z}[x]$, almost all monodromy groups of Belyĭ maps are either S_n or A_n, in agreement with what is to be expected based on Theorem 5.1.

Conjecture. $\Pr(\operatorname{Mon} \beta \in \{S_{\deg \beta}, A_{\deg \beta}\})$ *is a negligible function of* $\deg \beta$.

3.4. Forging $\beta\alpha$

In order to forge $\beta\alpha$ according to the verification algorithm, one must respond with a rational function f defined over the same field as $\beta\alpha$ so that `Monodromy` returns $\Delta_{\beta\alpha}$ and `Is`$S_n A_n$ finds sufficiently small probability that $\operatorname{Mon} f \in \{S_n, A_n\}$. Realistically, this implies that an adversary must produce a rational function f which approximates

Table 5.1.: Frequency of occurence of $S_{\deg\beta}$ and $A_{\deg\beta}$ as $\operatorname{Mon}\beta$

$\deg\beta$	$S_{\deg\beta}$	$A_{\deg\beta}$	Other
1	1.0000 (1)		
2	1.0000 (1)		
3	0.3333 (1)	0.6667 (2)	
4	0.2500 (2)	0.2500 (2)	0.5000 (4)
5	0.4444 (12)	0.2222 (6)	0.3333 (9)
6	0.4564 (68)	0.2282 (34)	0.3154 (47)
7	0.6699 (619)	0.2749 (254)	0.0552 (51)
8	0.6621 (4781)	0.2612 (1886)	0.0767 (554)
9	0.6991 (44189)	0.2895 (18295)	0.0114 (720)
10	0.7056 (440188)	0.2857 (178219)	0.0087 (5457)
11	0.7168 (4859273)	0.2829 (1918034)	0.0003 (1895)
12	0.7212 (58217055)	0.2764 (22311437)	0.0024 (192390)

a Belyĭ map equivalent to $\beta\alpha$ which satisfies $\operatorname{Mon} f \notin \{S_n, A_n\}$. As the probability of the latter is can be made arbitrarily small by choosing $\deg\alpha$ sufficiently large, the probability of finding such as f can also be made arbitrarily small.

4. The Protocol

By abuse of notation, for a Belyĭ map β of genus zero, the constellation Δ_β is considered to consist of both the constellation and the extending pattern.

4.1. Setup

Let h be a cryptographic hash function and let λ be a security parameter (see Section 1.5.). As part of the setup for the protocol, assume that a table of dynamical Belyĭ maps of degree $n_1(\lambda) \leq n \leq n_2(\lambda)$ and their extending patterns has been computed and made public. Further assume that a key generation algorithm has been run and assigned to each party i a private key which is a Belyĭ map γ_i on an elliptic cuve E_i, both defined over the number field $\mathbb{Q}(\alpha_i)$. The corresponding public key consists of the algebraic number α_i, the elliptic curve E_i, and the constellation Δ_{γ_i}.

4.2. Identification Scheme

In initiating the scheme, party i selects at random a dynamical Belyĭ map ρ_i from the public table of dynamical Belyĭ maps. Party i then sends the message $\left(\rho_i, \Delta_{\rho_i}\right)$ to party j. Party j randomly selects a dynamical Belyĭ map ρ_j from the table and sends the message $\left(\rho_j, \Delta_{\rho_j}, \rho_j\rho_i\gamma_j\right)$ to i. Finally, i sends the message $\left(\rho_i\rho_j\gamma_i\right)$. At the end of the second and third flows, the receiving party verifies the message as described below and either accepts and proceeds with the protocol, or rejects and terminates the protocol.

$$\begin{tikzcd}
i \arrow[r, shift left=6, "\rho_i, \Delta_{\rho_i}"] \arrow[r, leftarrow, "\rho_j, \Delta_{\rho_j}, \rho_j\rho_i\gamma_j"] \arrow[r, shift right=6, "\rho_i\rho_j\gamma_i"] & j
\end{tikzcd}$$

This flow pattern is inspired by that of [9], where composition with γ replaces the use of a signature.

4.3. Verification

Let $\beta := \rho_a\rho_b\gamma_a$. The verifier begins by computing the constellation Δ_β, both numerically using `Monodromy` and by composing $\Delta_{\rho_a}\circ\Delta_{\rho_b}\circ\Delta_{\gamma_a}$ using `ComposeConstellations`. If both methods yield the same result for Δ_β, the verifier uses `Is`S_nA_n to determine whether Mon β has sufficiently small probability of being S_n or A_n. If this is the case, the verifier accepts the identity of the prover.

4.4. Practicalities

All of the methods used during the verifcation step of the protocol are efficient. The most time-consuming aspect of the verification is the numerical computation of the monodromy. Indeed, although `Monodromy` can compute the monodromy of a genus one Belyĭ map with hundreds of edges in just a few seconds, the Belyĭ maps required for this protocol would be significantly larger due to the multiplicativity of degree under composition.

For its use in cryptography, one might hope that computation of a sufficiently complicated Belyĭ map from its associated constellation is quantum-resistant. The conventional method for computing a Belyĭ map from a dessin is through the use of Gröbner bases, which, in the worst case, have doubly exponential complexity [43]. Cryptographic protocols based on systems of multivariate quadratic polynomials also rely on the algorithmic complexity of Gröbner bases, which provide a method for breaking these systems [7]. The fact that such systems are considered quantum-resistant gives some indication that computing Belyĭ maps may be a prohibitive task for a quantum computer. Though there has been some recent progress on computing Belyĭ maps using modular forms [33], the problem still seems to be intractible for moderately complex examples.

Conjecture. *For sufficiently large n, g, and d, computing a Belyĭ map of degree n and genus g over a number field of degree d is quantum-resistant.*

Unfortunately, the degrees of the composite Belyĭ maps involved in the protocol must be very large in order to obtain sufficient randomness for γ_i and ρ_i. As the degrees of both the composite Belyĭ maps and the fields of definition are multiplicative, the quantity of coefficients involved in an exact representation of $\rho_i\rho_j\gamma_i$ is much too large to be useful for a cryptographic protocol. If it were possible to devise a method which could verify the monodromy of a Belyĭ map given its image in a finite field or a truncated polynomial ring, or both, this would alleviate the memory and communication burden resulting from the use of the complete Belyĭ map.

REFERENCES

[1] C. Alonso, J. Gutierrez, and T. Recio. A rational function decomposition algorithm by near-separated polynomials. *J. Symbolic Comput.*, 19(6):527–544, 1995.

[2] A. O. L. Atkin and H. P. F. Swinnerton-Dyer. Modular forms on noncongruence subgroups. pages 1–25, 1971.

[3] M. Ayad and P. Fleischman. On the decomposition of rational functions. *J. Symbolic Comput.*, 43(4):259–274, 2008.

[4] D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. *Numerically solving polynomial systems with Bertini*, volume 25 of *Software, Environments, and Tools*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2013.

[5] G. V. Belyĭ. On Galois extensions of a maximal cyclotomic field. *Math. USSR Izv.*, 14(2):247–256, 1980.

[6] G. V. Belyĭ. Another proof of the three points theorem. *Sb. Math.*, 193(3):329–332, 2002.

[7] L. Bettale, J.-C. Faugère, and L. Perret. Hybrid approach for solving multivariate systems over finite fields. *J. Math. Cryptol.*, 3(3):177–197, 2009.

[8] B. Birch. Noncongruence subgroups, covers and drawings. In L. Schneps, editor, *The Grothendieck Theory of Dessins d'Enfants*, number 200 in London Mathematical Society Lecture Note Series, pages 25–46. Cambridge University Press, 1994.

[9] S. Blake-Wilson and A. Menezes. Entity authentication and authenticated key transport protocols employing asymmetric techniques. In B. Christianson, B. Crispo, M. Lomas, and M. Roe, editors, *Security Protocols*, number 1361 in Lecture Notes in Computer Science, pages 137–158. Springer-Verlag, 1998.

[10] G. Boccara. Nombre de representations d'une permutation comme produit de deux cycles de longueurs donnees. *Discrete Mathematics*, 29(2):105–134, 1980.

[11] D. Bump. *Lie Groups*. Number 225 in Graduate Texts in Mathematics. Springer-Verlag, second edition, 2013.

[12] L. Cangelmi. Factorizations of an n-cycle into two n-cycles. *Europ. J. Combinatorics*, 24(7):849–853, 2003.

[13] K. Conrad. Recognizing Galois groups S_n and A_n. `https://www.math.uconn.edu/~kconrad/blurbs/galoistheory/galoisSnAn.pdf`.

[14] J.-M. Couveignes and L. Granboulan. *Dessins from a geometric point of view*, pages 79–113. Number 200 in London Mathematical Society Lecture Note Series. Cambridge University Press, 1994.

[15] J. H. Davenport and G. C. Smith. Fast recognition of alternating and symmetric Galois groups. *J. Pure Appl. Algebra*, 153(1):17–25, 2000.

[16] W. Decker, G.-M. Greuel, G. Pfister, and H. Schönemann. Singular 4-1-1 — A computer algebra system for polynomial computations. `http://www.singular.uni-kl.de`, 2018.

[17] B. Deconinck and M. van Hoeij. Computing Riemann matrices of algebraic curves. *Phys. D*, 152/153:28–46, 2001. Advances in nonlinear mathematics and science.

[18] R. Dietmann. On the distribution of Galois groups. *Mathematika*, 58(1):35–44, 2012.

[19] J. D. Dixon and B. Mortimer. *Permutation Groups.* Number 163 in Graduate Texts in Mathematics. Springer-Verlag, 1996.

[20] D. S. Dummit and R. M. Foote. *Abstract Algebra.* John Wiley & Sons, 3rd edition, 2004.

[21] P. X. Gallagher. The large sieve and probabilistic Galois theory. In *Analytic Number Theory*, volume XXIV of *Proceedings of the Symposium in Pure Mathematics of the American Mathematical Society.* American Mathematical Society, 1973.

[22] GAP – Groups, Algorithms, and Programming, Version 4.8.7. `https://www.gap-system.org`, 2017.

[23] E. H. Goins. Introduction to Dessins d'Enfants. `https://www.math.purdue.edu/~egoins/notes/dessin_denfants.pdf`.

[24] J. Gutierrez, R. Rubio, and D. Sevilla. On multivariate rational function decomposition. *J. Symbolic Comput.*, 33(5):545–562, 2002.

[25] J. Gutierrez and D. Sevilla. Computation of unirational fields. *J. Symbolic Comput.*, 41(11):1222–1244, 2006.

[26] J. Harris. Galois groups of enumerative problems. *Duke Math. J.*, 46(4):685–724, 1979.

[27] J. Heintz. On polynomials with symmetric Galois group which are easy to compute. *Theoret. Comput. Sci.*, 47(1):99–105, 1986.

[28] G. James and M. Liebeck. *Representations and Characters of Groups.* Cambridge University Press, second edition, 2001.

[29] A. Javanpeykar. Polynomial bounds for Arakelov invariants of Belyi curves. *Algebra Number Theory*, 8(1):89–140, 2014. With an appendix by Peter Bruin.

[30] G. A. Jones and M. Streit. Galois groups, monodromy groups and cartographic groups. In *Geometric Galois actions, 2*, volume 243 of *London Mathematical Society Lecture Note Series*, pages 25–65. Cambridge University Press, 1997.

[31] R. Kannan, A. K. Lenstra, and L. Lovàsz. Polynomial factorization and nonrandomness of bits of algebraic and some transcendental numbers. *Math. Comp.*, 50(181):235–250, 1988.

[32] J. Katz and Y. Lindell. *Introduction to Modern Cryptography.* CRC Press, second edition, 2015.

[33] M. Klug, M. Musty, S. Schiavone, and J. Voight. Numerical calculuation of three-point branched covers of the projective line. *LMS J. Comput. Math.*, 17(1):379–430, 2014.

[34] B. Köck. Belyi's theorem revisited. *Beiträge Algebra Geom.*, 45(1):253–265, 2004.

[35] S. Kohl. RCWA, residue-class-wise affine groups, Version 4.5.1. `https://stefan-kohl.github.io/rcwa.html`, Mar 2017. Refereed GAP package.

[36] A. Kondratyev, H. Stetter, and F. Winkler. Numerical Computation of Gröbner Bases. In V. Ghanza, E. Mayr, and E. Vorozhtov, editors, *Proc. 7th Workshop on Computer Algebra in Scientific Computing (CASC-2004)*, pages 295–306. Technische Univ. Muenchen, 2004.

[37] D. Krumm and N. Sutherland. Galois groups over rational function fields and explicit Hilbert irreducibility. 2017.

[38] S. K. Lando and A. K. Zvonkin. *Graphs on Surfaces and Their Applications.* Number 141 in Encyclopaedia of Mathematical Sciences. Springer-Verlag, 2004.

[39] S. Lang. *Algebra.* Number 211 in Graduate Texts in Mathematics. Springer-Verlag, 2002.

[40] A. K. Lenstra, J. Hendrik W. Lenstra, and L. Lovàsz. Factoring polynomials with rational coefficients. *Math. Ann.*, 261(4):515–534, 1982.

[41] W. S. Massey. *A Basic Course in Algebraic Topology.* Number 127 in Graduate Texts in Mathematics. Springer-Verlag, 1991.

[42] M. Matsumoto. Arithmetic fundamental groups and moduli of curves. In *School on Algebraic Geometry (Trieste, 1999)*, volume 1 of *ICTP Lect. Notes*, pages 355–383. Abdus Salam Int. Cent. Theoret. Phys., Trieste, 2000.

[43] E. W. Mayr and S. Ritscher. Dimension-dependent bounds for Gröbner bases of polynomial ideals. *J. Symbolic Comput.*, 49:78–94, 2013.

[44] B. McKay. The simultaneous conjugacy problem in the symmetric group s_n. MathOverflow. `https://mathoverflow.net/q/162453`.

[45] A. D. Mednykh. Nonequivalent coverings of Riemann surfaces with a prescribed ramification type. *Siberian Math. J.*, 25(4):606–625, 1984.

[46] A. D. Mednykh. Branched coverings of Riemann surfaces whose branch orders coincide with the multiplicity. *Comm. Algebra*, 18(5):1517–1533, 1990.

[47] R. Miranda. *Algebraic Curves and Riemann Surfaces.* Number 5 in Graduate studies in mathematics. American Mathematical Society, 1995.

[48] J. R. Munkres. *Topology.* Prentice Hall, Inc., second edition, 2000.

[49] R. Remmert. *Theory of Complex Functions.* Number 122 in Graduate Texts in Mathematics. Springer-Verlag, 1991.

[50] X.-F. Roblot. Polynomial factorization algorithms over number fields. *J. Symbolic Comput.*, 38(5):1429–1443, 2004.

[51] J. J. Rotman. *An Introduction to Algebraic Topology.* Number 119 in Graduate Texts in Mathematics. Springer-Verlag, 1988.

[52] J. J. Rotman. *An Introduction to Homological Algebra.* Universitext. Springer-Verlag, second edition, 2009.

[53] SageMath, the Sage Mathematics Software System (Version 8.1), 2017. `https://www.sagemath.org`.

[54] V. Schemmel. Ueber relative primzahlen. *J. Reine Angew. Math.*, 70:191–192, 1869.

[55] L. Schneps. Dessins d'enfants on the riemann sphere. In L. Schneps, editor, *The Grothendieck Theory of Dessins d'Enfants*, number 200 in London Mathematical Society Lecure Note Series, pages 47–77. Cambridge University Press, 1994.

[56] J.-P. Serre. *Topics in Galois Theory.* Number 1 in Research Notes in Mathematics. A K Peters, 2 edition, 2007.

[57] J. Sijsling and J. Voight. On computing Belyi maps. In *Numéro consacré au trimestre: Méthodes arithmétiques et applications, automne 2013*, volume 2014/1 of *Publ. Math. Besançon Algèbre Theéorie Nr.*, pages 73–131. 2014.

[58] E. H. Spanier. *Algebraic Topology.* McGraw-Hill Series in Higher Mathematics. McGraw-Hill, 1966.

[59] V. Strassen. Polynomials with rational coefficients which are hard to compute. *SIAM J. Comput.*, 3:128–149, 1974.

[60] M. Suzuki. *Group Theory I.* Number 247 in Grundlehren der matematischen Wissenschaften. Springer-Verlag, 1982.

[61] T. tom Dieck. *Algebraic Topology.* EMS Textbooks in Mathematics. European Mathematical Society, 2008.

[62] G. Turnwald. Some notes on monodromy groups of polynomials. In K. Győry, H. Iwaniec, and J. Urbanowicz, editors, *Number Theory in Progress*, volume 1. de Gruyter, 1999.

[63] B. L. van der Waerden. Die Seltenheit der Gleichungen mit Affekt. *Math. Ann.*, 109(1):13–16, 1934.

[64] B. L. van der Waerden. Die Seltenheit der reduziblen Gleichungen und der Gleichungen mit Affekt. *Monatsh. Math. Phys.*, 43(1):133–147, 1936.

[65] M. van Hoeij. Roadmap algorithm for constructing dessins d'enfants. 2016.

[66] M. van Hoeij and V. J. Kunwar. Classifying (near)-Belyi maps with five exceptional point, 2016.

[67] Wolfram Research, Inc. Mathematica, Version 11.3. Champaign, IL, 2018.

[68] M. M. Wood. Belyi-extending maps and the Galois action on dessins d'enfants. *Publ. Res. Inst. Math. Sci.*, 42(3):721–737, 2006.

[69] R. Zippel. *Effective Polynomial Computation.* Number 241 in The Springer International Series in Engineering and Computer Science. Springer-Verlag, 1993.