

# **Computational Riemann surfaces**

*Marko Puza*

Year 4 Project  
School of Mathematics  
University of Edinburgh  
2018

# Abstract

The main purpose of the project is to develop a computational tool, CyclePainter, (to be used within Sage) for working with closed paths on algebraic Riemann surfaces. Initially, a self-contained relevant theory of Riemann surfaces is presented, including definitions of the various concepts that are necessary for understanding CyclePainter, such as genus, the monodromy group, the homology group and the Riemann matrix. A thorough description of the implementation and functionality of CyclePainter follows. Finally, motivation on how this tool can be useful in the study of Riemann surfaces with symmetries is given, as well as a selection of examples.

This project report is submitted in partial fulfilment of the requirements for the degree of *BSc. Computer science & Mathematics*.

# Contents

<b>Abstract</b>	<b>2</b>
<b>1 Introduction</b>	<b>4</b>
1.1 Riemann surfaces . . . . .	5
1.2 Hyperelliptic curves . . . . .	7
1.3 Holomorphic/meromorphic maps . . . . .	8
1.4 Topology on Riemann surfaces . . . . .	12
1.5 Homology . . . . .	16
1.6 Monodromy . . . . .	21
1.7 Holomorphic differentials & the period matrix . . . . .	23
1.8 Riemann surfaces with symmetries . . . . .	25
<b>2 CyclePainter</b>	<b>27</b>
2.1 Riemann surfaces in mathematics software systems . . . . .	27
2.2 Crucial algorithms available . . . . .	28
2.3 Functionality . . . . .	30
2.4 Calculating the point $p_c$ . . . . .	34
2.5 Modifying the monodromy . . . . .	35
2.6 Application of automorphisms . . . . .	41
<b>3 Examples</b>	<b>43</b>
3.1 A simple hyperelliptic curve . . . . .	43
3.2 Curve $x^2 = y^6 + 5y^3 + 1$ . . . . .	44
3.3 Klein's curve . . . . .	47
<b>4 Conclusion</b>	<b>51</b>

# Chapter 1

## Introduction

Riemann surfaces are an important meeting point of complex analysis, algebra, geometry and topology. In the recent years, with the advance of computers, computational tools and algorithms for dealing with Riemann surfaces have been also developed, for example to construct homology bases or for computing the surface's associated period matrix. The importance of developing such tools is motivated at the very least by their omnipresence in the mathematical sciences. Riemann surfaces naturally appear as elliptic curves over the complex numbers, as the worldsheets of strings in the string theory; are in 1-1 correspondence with hyperbolic surfaces, and feature in elliptic-curve cryptography which utilizes elliptic curves over finite fields. The motivation that has driven many of the authors of the above mentioned computational tools is also, more particularly, the theory of periodic and quasiperiodic solutions of integrable partial differential equations. One of the computational tools for Riemann surfaces, that has been developed as part of the PhD thesis of Tim Northover some years ago, was CyclePainter. This tool for Maple, found useful by some researchers, has not been maintained since. Subsequently, we will describe its resurrection.

In this chapter we will develop a working theoretical understanding of Riemann surfaces and derive in detail their topological and geometric properties. This introduction will be self-contained. As is common within the field of complex analysis, we will see that imposing a complex structure on manifolds results in a rich class of “nice” properties. Indeed, we will see the topological structure embedded in Riemann surfaces which allows one to employ their intuition in order to understand them. The main objects of our interest will be compact Riemann surfaces and much of the theory will converge towards the central idea: how the maps between compact Riemann surfaces can be naturally understood as sheeted coverings.

The definitions, theorems and narrative of the theoretical introduction is mostly based on the book “Elliptic curves” by McKean & Moll [9], the instructive lectures notes by Bobenko [1] and lecture notes by Teleman [6]. There was an attempt to compile information on the topic in the most comprehensive way.

## Conventions and own work

Throughout the whole chapter, by *neighbourhoods* we will always mean *open neighbourhoods*.

For whom it may concern, I include a list of arguments in the theoretical first chapter that have been devised, modified or augmented by myself. These are: proofs of Theorems 3, 12, 18, 25; parts of the proof of Theorem 8 and section 1.6 on monodromy.

### 1.1 Riemann surfaces

Riemann surfaces are manifolds of (real) dimension 2, endowed with an additional complex structure. It is a manifestation of the complex structure that Riemann surfaces are *oriented*. Intuitively, at a small scale, they look like the open unit disk in the complex plane.

**Definition 1.** A *Riemann surface*  $\mathcal{M}$  is a connected complex manifold of (complex) dimension 1. This involves three things:

1.  $\mathcal{M}$  is a topological space which admits a countable open cover  $\{U_\alpha\}_{\alpha \in A}$ .
2. Each  $U_\alpha$  is equipped with a *local coordinate chart*, that is, a homeomorphism  $z_\alpha : U_\alpha \rightarrow D$  to the open unit disk  $D \subset \mathbb{C}$ .
3. On any non-empty intersection  $U_\alpha \cap U_\beta$ , the *transition function*

$$z_\beta \circ z_\alpha^{-1} : z_\alpha(U_\alpha \cap U_\beta) \rightarrow z_\beta(U_\alpha \cap U_\beta)$$

is holomorphic.

The collection  $\{U_\alpha, z_\alpha\}_{\alpha \in A}$  is called a *complex atlas*. The maximal atlas<sup>1</sup> will be called a *complex structure* (often denoted as  $\Sigma$ , or  $\Sigma_{\mathcal{M}}$ ). We will say that  $\mathcal{M}$  is *compact* if it is compact as a topological space.

In the following text, we will be mostly concerned with compact Riemann surfaces. It is a crucial result, which we will not prove here, that *every Riemann surface that is compact is also algebraic*; that is, it is the normalized and compactified zero-set of some polynomial  $P(z, w)$  over the complex number field. In CyclePainter, Riemann surfaces will be specified by their polynomials.

The concept of Riemann surfaces can be (for example) justified by forming a tool for dealing with multi-valuedness of complex functions. The usual approach is to substitute the domain of any such function by a corresponding Riemann surface; making it effectively single-valued. The classical example is that of the complex square root. We may define the complex square root be the  $w$ -projection from  $\mathcal{S}$  to  $\mathbb{C}$  where  $\mathcal{S}$  is the Riemann surface given by:

$$\mathcal{S} = \{(z, w) \in \mathbb{C}^2 \mid w^2 = z\}$$

---

<sup>1</sup>This can be understood as the atlas which contains all possible coordinate charts compatible with the given constraints.

## Non-singular algebraic curves as Riemann surfaces

We will start by giving a whole class of examples. The next subsection will show how to equip non-singular algebraic curves with a complex structure, allowing us to view them as Riemann surfaces.

**Definition 2.** Let  $P \in \mathbb{C}[z, w]$  be an irreducible polynomial. A *non-singular algebraic curve*  $\mathcal{C}$  is:

$$\mathcal{C} = P^{-1}(0) = \{(\mu, \lambda) \mid P(\mu, \lambda) = 0\} \subset \mathbb{C}^2$$

with the *non-singularity* condition being:

$$\nabla P|_{P(\mu, \lambda)=0} \neq 0$$

In words, the gradient does not vanish at any point of the curve. Any point at which the gradient vanishes will be called *singular point*.

To introduce a complex structure on the zero-set  $P^{-1}(0)$ , we will use a simplified version of the Analytic Implicit Function Theorem. The below derivation is due to [6].

Consider an arbitrary point  $(\mu, \lambda) \in P^{-1}(0)$  and break  $z, w, P(z, w)$  into real and imaginary parts:

$$z = x + iy, \quad w = u + iv, \quad P(z, w) = M(z, w) + iN(z, w)$$

The Jacobian of  $P$  is then:

$$J_P = \begin{pmatrix} \frac{\partial M}{\partial x} & \frac{\partial M}{\partial y} & \frac{\partial M}{\partial u} & \frac{\partial M}{\partial v} \\ \frac{\partial N}{\partial x} & \frac{\partial N}{\partial y} & \frac{\partial N}{\partial u} & \frac{\partial N}{\partial v} \end{pmatrix} = \begin{pmatrix} \frac{\partial M}{\partial x} & \frac{\partial M}{\partial y} & \frac{\partial M}{\partial u} & \frac{\partial M}{\partial v} \\ -\frac{\partial M}{\partial y} & \frac{\partial M}{\partial x} & -\frac{\partial M}{\partial v} & \frac{\partial M}{\partial u} \end{pmatrix}$$

where the latter equality comes from  $P$  being a polynomial and thus analytic, and applying the Cauchy-Riemann equations. As  $P$  is non-singular, we have either  $\frac{\partial P}{\partial z}|_{(\mu, \lambda)} \neq 0$  or  $\frac{\partial P}{\partial w}|_{(\mu, \lambda)} \neq 0$ . Without loss of generality, suppose it is the latter. Then by the Cauchy-Riemann equations either  $\frac{\partial M}{\partial u}$  or  $\frac{\partial N}{\partial u} = -\frac{\partial M}{\partial v}$  is non-zero. The matrix given by

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ & & J_P & \end{pmatrix}$$

has a full rank, and thus is *in the vicinity of*  $(\mu, \lambda)$  a change-of-variables matrix of  $(x, y, u, v) \rightarrow (x, y, M, N)$ . By the analytic version of the inverse function theorem, for when  $P(z, w) = M(x, y, u, v) = N(x, y, u, v) = 0$ , there are smooth functions  $u(x, y)$  and  $v(x, y)$  defining together (one can check that they satisfy the Cauchy-Riemann equations) a holomorphic function  $w = w(z)$ . We can therefore take  $z$  as a local coordinate here.

At any point  $(\mu, \lambda)$  where both  $\frac{\partial P}{\partial z}|_{(\mu, \lambda)} \neq 0$  and  $\frac{\partial P}{\partial w}|_{(\mu, \lambda)} \neq 0$ , we will take either  $z$  or  $w$  as a local coordinate. Finally, we may perform compactification of the curve to obtain a compact Riemann surface.

## 1.2 Hyperelliptic curves

An important subclass of algebraic Riemann surfaces are *hyperelliptic curves*. These are defined by algebraic polynomial equations of the form:

$$w^2 = f(z) \text{ where } f(z) = c \prod_{i=1}^n (z - z_i), \quad c \neq 0, \quad n \geq 3$$

where, without loss of generality  $c$  is usually taken to be scaled to  $c = 1$ . For the cases where  $f$  is a polynomial of degree  $n = 3$  or  $n = 4$ , the curve is called *elliptic*.

**Theorem 3.** *A hyperelliptic curve is non-singular if and only if  $f(z)$  has no repeated roots.*

*Proof.* ( $\Leftarrow$ ) Assume that the hyperelliptic curve has no repeated roots and suppose for contradiction that it is singular. Then, there must be  $(\mu_0, \lambda_0)$  on the curve for which  $\frac{\partial(w^2 - f(z))}{\partial w}|_{(\mu_0, \lambda_0)} = \frac{\partial(w^2 - f(z))}{\partial z}|_{(\mu_0, \lambda_0)} = 0$ . This is equivalent to  $(\mu_0, \lambda_0)$  satisfying simultaneously equations:

$$\lambda_0^2 = f(\mu_0) \tag{1.1}$$

$$2\lambda_0 = 0 \tag{1.2}$$

$$\left. \frac{\partial(w^2 - f(z))}{\partial z} \right|_{(\mu_0, \lambda_0)} = -f'(\mu_0) = 0 \tag{1.3}$$

Using (1.2) in (1.1), we see that  $\mu_0$  is a root of  $f$ . Since it has no repeated roots, we can write  $f(z) = (z - \mu_0)g(z)$  where  $g(\mu_0) \neq 0$ . But then  $f'(z) = g(z) + (z - \mu_0)g'(z)$  and  $f'(\mu_0) = g(\mu_0) \neq 0$ , which contradicts (1.3).

( $\Rightarrow$ ) Let  $\mu_0$  be a root of  $f$  with multiplicity  $> 1$ . Then  $(\mu_0, 0)$  satisfies all (1.1), (1.2) and (1.3) and hence constitutes a singular point on the curve.  $\square$

Hyperelliptic curve have two sheets  $w_{\pm} = \pm\sqrt{f(z)}$  and are compactified by adding  $\{\infty\}$  in case  $n$  is odd, or  $\{\infty^{\pm}\}$  in case  $n$  is even.

## Singularities, Puiseux expansions

There are also Riemann surfaces associated with singular algebraic curves. Consider a surface  $\mathcal{S}$  which possesses singular points. It can be shown that the singularities (of analytic sets) are isolated. For such a surface, there exists a Riemann surface  $\mathcal{R}$  to which we can biholomorphically map all of the regular points, effectively replacing the singularities of  $\mathcal{S}$  by smooth points. In the customary terminology, this process is called *normalization* and it is said that  $\mathcal{R}$  *resolves* the singularities of  $\mathcal{S}$ .

The theory of normalization is out of the scope of this project and the only thing

that we will take out of this discussion is that normalization can be done; and it can be done algorithmically via Puiseux expansions<sup>2</sup>.

### 1.3 Holomorphic/meromorphic maps

Let us now characterize the maps between Riemann surfaces.

**Definition 4** (Holomorphic mapping). A mapping  $f : \mathcal{M} \rightarrow \mathcal{N}$  between Riemann surfaces is called *holomorphic* if the complex structures of the surfaces are holomorphically compatible under the map. That is, for all  $(U, z) \in \Sigma_{\mathcal{M}}, (V, w) \in \Sigma_{\mathcal{N}}$ , whenever we have  $U \cap f^{-1}(V)$ , then the composite mapping

$$w \circ f \circ z^{-1} : \mathbb{C} \supset z(U \cap f^{-1}(V)) \longrightarrow w(V) \subset \mathbb{C}$$

is holomorphic.

$$\begin{array}{ccc} U & \xrightarrow{f} & V \\ \downarrow z & & \downarrow w \\ \mathbb{C} & & \mathbb{C} \end{array}$$

**Definition 5** (Holomorphic/meromorphic function). We call a holomorphic mapping to  $\mathbb{C}$  a *holomorphic function*. We call a holomorphic mapping to  $\widehat{\mathbb{C}}$ , the Riemann sphere, a *meromorphic* function.

It is a direct consequence of the above definitions that any holomorphic mapping  $f : \mathcal{M} \rightarrow \mathcal{N}$  takes holomorphic functions on  $\mathcal{N}$  to holomorphic functions on  $\mathcal{M}$ . Let  $h : \mathcal{N} \rightarrow \mathbb{C}$  be a holomorphic function on  $\mathcal{N}$ . But then  $h \circ f : f^{-1}(\mathcal{N}) \rightarrow \mathbb{C}$  is a holomorphic mapping to  $\mathbb{C}$  (because  $f$  being a holomorphic mapping implies any  $z_{\mathcal{M}} \circ f \circ z_{\mathcal{N}}^{-1}$  holomorphic,  $h$  being a holomorphic function implies any  $z_{\mathbb{C}} \circ h \circ z_{\mathcal{N}}^{-1}$  holomorphic, and thus the composition of these,  $z_{\mathbb{C}} \circ h \circ f \circ z_{\mathcal{N}}^{-1}$  is also holomorphic) and thus a holomorphic function on  $\mathcal{M}$ .

The theorem of the next subsection describes the local behaviour of holomorphic mappings and will be important to develop the theory of Riemann surfaces. It justifies the (vague) concept of *sheets* on the surface that we will need. To understand its proof, we first need to introduce the following definitions from general complex variable analysis:

**Definition 6** ((Local) analytic isomorphism). Let  $U \subset \mathbb{C}$  be an open set. An analytic function  $f : U \rightarrow \mathbb{C}$  is called an *analytic isomorphism* if there exists an analytic function  $g$  defined on  $f(U)$  such that  $g(f(z)) = z$  for all  $z \in U$ .

We say that  $f$  is a *local analytic isomorphism* at  $z_0$  if there exists an open set  $U \ni z_0$  such that  $f$  is an analytic isomorphism on  $U$ .

And we also need the following theorem, proof of which we leave omitted.

**Theorem 7** (Complex Inverse Function Theorem). *If  $f$  is holomorphic at  $z_0$  and  $f'(z_0) \neq 0$ , then  $f$  is a local analytic isomorphism at  $z_0$ .*

---

<sup>2</sup>Puiseux expansions can be informally understood as Laurent series with fractional exponents whose denominator is bounded.

## “Holomorphic mappings are locally like the map $x \mapsto x^k$ ”

**Theorem 8** (Local form of holomorphic mappings). *Let  $f : \mathcal{M} \rightarrow \mathcal{N}$  be a holomorphic mapping which is non-constant near  $m \in \mathcal{M}$ . Denote  $n = f(m)$ . Consider a given  $(V, z_{\mathcal{N}}) \in \Sigma_{\mathcal{N}}$  where  $V$  is a sufficiently small neighbourhood of  $n$ , with  $z_{\mathcal{N}}(n) = 0 \in D$  ( $D$  the unit complex disk). Then there exists  $(U, z_{\mathcal{M}}) \in \Sigma_{\mathcal{M}}$  with the properties:*

- $U$  is a neighbourhood of  $m$
- $f(U) \subseteq V$
- $(z_{\mathcal{N}} \circ f \circ z_{\mathcal{M}}^{-1})(x) = x^k$  for all  $x \in U$ , where  $k \in \mathbb{N}$ .

$$\begin{array}{ccc} U & \xrightarrow{f} & V \\ \downarrow z_{\mathcal{M}} & & \downarrow z_{\mathcal{N}} \\ D & \xrightarrow{x \mapsto x^k} & D \end{array}$$

That is, the above diagram commutes.

*Proof.* Given some  $(V', z_{\mathcal{N}}) \in \Sigma_{\mathcal{N}}$ , choose  $(U', z') \in \Sigma_{\mathcal{M}}$  centered at  $m$ . Since  $f$  is a holomorphic mapping, we have that  $g := z_{\mathcal{N}} \circ f \circ z'^{-1}$  is a holomorphic/analytic function  $D \rightarrow D$  centred about 0. Let  $k$  be the order of zero of  $g$  at 0:

$$g(x) = g_k x^k + \mathcal{O}(x^{k+1}) \quad \text{where } g_k \neq 0$$

Using this, we may define near 0 the analytic  $k$ -th root of  $g$  as:

$$g^{\frac{1}{k}}(x) = g_k^{\frac{1}{k}} x + \mathcal{O}(x^2)$$

which is, by the Complex Inverse Function Theorem a local analytic isomorphism at 0 (since  $\partial_x g^{\frac{1}{k}}(0) = g_k^{\frac{1}{k}} \neq 0$ ). But then,  $g^{\frac{1}{k}} \circ z'$  is a local analytic isomorphism sending some neighbourhood  $U$  of  $m \in U'$  to a neighbourhood  $z_{\mathcal{N}}(V)$  (for some<sup>3</sup>  $V$ ) of 0  $\subset \mathbb{C}$ , whose  $n$ -th power is  $g \circ z' = z_{\mathcal{N}} \circ f$  (the last equality from  $g = z_{\mathcal{N}} \circ f \circ z'^{-1}$ ). We can choose

$$z_{\mathcal{M}} := g^{\frac{1}{k}} \circ z'$$

which has the property:

$$(z_{\mathcal{N}} \circ f)(x) = z_{\mathcal{M}}(x)^k$$

and the claim is proved.  $\square$

It is a corollary of the local form of holomorphic mappings that  $f$  is an open map, as it can be (locally) written as a composition of open maps. To test the power with which the above theorem equips us, we will now address a Riemann surface version of the classical Louiville theorem:

---

<sup>3</sup>This is the “sufficiently small”  $V$  from the theorem statement.

**Theorem 9.** *On any compact Riemann surface there are no non-constant holomorphic functions.*

*Proof.* Suppose  $\mathcal{M}$  is a compact Riemann surface and that  $f : \mathcal{M} \rightarrow \mathbb{C}$  is a holomorphic function. Then, since  $f$  is both open and continuous, it sends compact sets to compact sets. In particular,  $f(\mathcal{M})$  is compact (thus closed) and open. Moreover,  $\mathcal{M}$  is connected and so will be  $f(\mathcal{M})$ . But then  $f(\mathcal{M})$  must be a compact connected component (because  $f(\mathcal{M})$  and  $\mathbb{C} \setminus f(\mathcal{M})$  are both open) of  $\mathbb{C}$ . There is, however, no compact connected component of  $\mathbb{C}$ .  $\square$

## Sheets of compact Riemann surfaces

Non-constant holomorphic mappings  $f : \mathcal{M} \rightarrow \mathcal{N}$  are also discrete. That is, for any point  $n \in \mathcal{N}$  and any point  $m \in \mathcal{M}$ , there is a neighbourhood  $U$  of  $m$  such that the cardinality of the set  $f^{-1}(n) \cap U$  is bounded by  $|f^{-1}(n) \cap U| \leq 1$ . If they weren't, an existence of a limiting point within  $f^{-1}(n)$ , and hence constancy of  $f$  would be implied. We will from now on refer to non-constant holomorphic mappings as *holomorphic coverings*.

**Definition 10** (Branch points). For a given holomorphic covering  $f : \mathcal{M} \rightarrow \mathcal{N}$ , we will call a *branch point of  $f$*  any point  $m \in \mathcal{M}$  which has no neighbourhood  $U$  with  $f|_U$  injective.

We note that it is a result of standard complex analysis that this condition is equivalent to:

$$\frac{\partial(z_{\mathcal{N}} \circ f \circ z_{\mathcal{M}}^{-1})}{\partial x} \Big|_{z_{\mathcal{M}}(m)} = 0$$

**Definition 11** (Branch number). The number  $k$  from Theorem 8 above is independent of the choice of local coordinate. It signifies that  $f$  takes on the same value exactly  $k$  times in the neighbourhood of  $m \in \mathcal{M}$ . We will define the *branch number of  $f$  at  $m$*  to be  $k - 1$ .

$$b_f(m) := k - 1$$

In particular, branch points of  $f$  are exactly the points for which  $b_f(m) > 0$ .

Importantly, the set of branch points  $B = \{m \in \mathcal{M} \mid b_f(m) > 0\}$  of a holomorphic covering is finite. Were it not so, there would be a limiting point in  $B$ , as  $B$  is an infinite subset of a compact set. But no such point can exist, as every branch point  $b \in B$  has a neighbourhood (as implied by the Local form of holomorphic mappings)  $U$  in which  $f$  behaves as  $x \mapsto x^k$ . This map is injective on  $x \neq 0$  and so will be  $f$  on  $U \setminus \{m\}$ . Therefore, as there can only be one branch point in  $U$ , no branch point can be a limiting point.

We will now see how the conceptual distinction of sheets in connection to compact Riemann surfaces is justified. It is important, however, that the (numbers of) sheets are not a property intrinsic to the surface itself. Rather, these are determined by the holomorphic mapping. Indeed, a particular surface can cover different surfaces by different holomorphic coverings with distinct numbers

of sheets.

The discrete and finite nature of the branch points of coverings will further allow us to view the sheets as being glued together at a few “special” points - the branch points.

**Theorem 12.** *Let  $f : \mathcal{M} \rightarrow \mathcal{N}$  be a holomorphic covering. Then there exists a fixed number  $d \in \mathbb{N}$  such that for all  $n \in \mathcal{N}$ :*

$$\sum_{m \in f^{-1}(n)} (b_f(m) + 1) = d$$

*The number  $d$  is the degree of the covering  $f$  and we will call such a covering  $d$ -sheeted.*

*Proof.* Let  $B$  be the set of branch points of the covering  $f : \mathcal{M} \rightarrow \mathcal{N}$ . Take a particular point  $n_{\text{base}} \in \mathcal{N} \setminus f(B)$  and let  $d := |f^{-1}(n_{\text{base}})|$ . Note that we must have  $d < \infty$ , otherwise  $f$  would have a limiting point and be constant.

We will first show that  $|f^{-1}(n)| = d$  for all  $n \in \mathcal{N} \setminus f(B)$ . Indeed, consider any simple path  $\gamma \subset \mathcal{N} \setminus f(B)$  between  $n_{\text{base}}$  and  $n$ . We have  $f^{-1}(\gamma)$  an union of paths in  $\mathcal{M}$ , and, in fact this must be a disjoint union, since any intersection of the components of  $f^{-1}(\gamma)$  would imply this intersection being a branch point, which cannot be the case by choice of  $\gamma$ . But then, there must be the same number of endpoints of paths in  $f^{-1}(\gamma)$  on one end, as there is on the other. That is,  $|f^{-1}(n)| = |f^{-1}(n_{\text{base}})| = d$ . Since for all such  $n$  we have  $b_f(m) = 0 \forall m \in f^{-1}(n)$ , the theorem is established for this case.

Consider now any point  $n \in f(B)$  and construct a path  $\gamma$  between  $n_{\text{base}}$  and  $n$  such that  $\gamma \cap f(B) = n$ . From the definition of branch number, we know that all  $m \in f^{-1}(n)$  take any particular value exactly  $b_f(m) + 1$  times in their neighbourhood. By the previous part, there are however precisely  $d$  preimages of any point on  $\gamma$  arbitrarily close to  $n$ . In other words,  $\sum_{m \in f^{-1}(n)} (b_f(m) + 1) = d$ .  $\square$

We can also depict holomorphic coverings diagrammatically. Any connected subset of  $f(\mathcal{M})$  which surjects on  $\mathcal{N}$  and whose inclusion on  $\mathcal{N}$  is injective, we will refer to as *sheet*. By the previous discussion, an intersection set of any two sheets is finite and discrete.

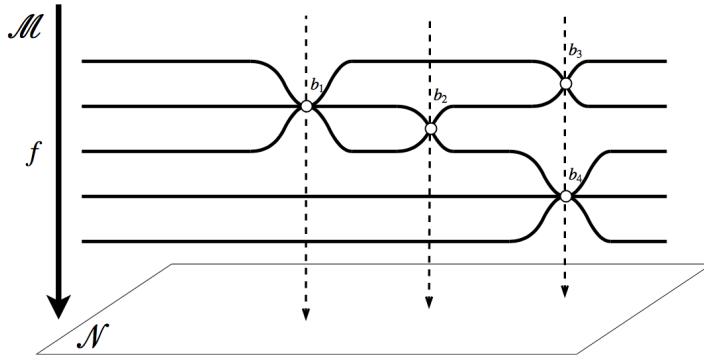


Figure 1.1: Branch points  $b_1, b_2, b_3, b_4$  of a covering which have branch numbers 3, 2, 2, 3 respectively.

### Riemann surfaces of algebraic curves as coverings

Finally, let us exemplify the concepts of the previous section in a more concrete setting of Riemann surfaces defined by algebraic curves. Let  $P \in \mathbb{C}[\lambda, \mu]$  be a defining polynomial of such a Riemann surface.

We can consider the projection  $(\lambda, \mu) \mapsto \lambda$  (or analogously,  $(\lambda, \mu) \mapsto \mu$ ) to be a holomorphic covering  $\pi_\lambda$ .

- $\pi_\lambda$  is a  $d$ -sheeted covering, where  $d$  is the degree of  $P$  as a polynomial in  $\mu$  (for a fixed  $\lambda$ , there are  $d$  solutions).
- The branch points of this coverings are (by the partial derivative characterization in the definition of branch points) the points with  $\frac{\partial P}{\partial \mu} = 0$ .

## 1.4 Topology on Riemann surfaces

The definition of Riemann surfaces allows us to view them as smooth oriented 2-dimensional real manifolds, which means that they bear a number of topological properties. Perhaps the most fundamental one is the genus of the surface, representing the number of holes that the surface has.

**Definition 13** (Genus). The *genus*  $g_M$  of a Riemann surface  $M$  is the (unique) number such that  $M$  is homeomorphic to a sphere with  $g_M$  handles.

We can also view genus as the maximum number of cuts along non-intersecting closed simple curves which do not disconnect the surface.

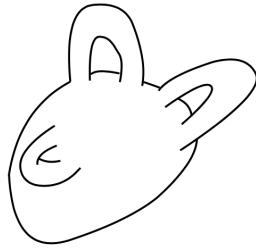


Figure 1.2: “Sphere” with 3 handles.

The fact that any Riemann surface *is* homeomorphic to a sphere with handles is not obvious and we omit the proof here. To formalize the vague term “sphere with handles”, we will understand this to be the following topological space.

**Definition 14** (Sphere with handles). Let  $P \subseteq \mathbb{R}^2$  be a regular  $4g$ -gon (containing the interior) with its *oriented* edges listed in counterclockwise order as  $\gamma_1, \gamma_2, \dots, \gamma_{4g}$ , where we will understand each  $\gamma_i$  as a parametrized path from  $\gamma_i(0)$  to  $\gamma_i(1)$ . The *sphere with  $g$  handles* is  $P / \sim$ , where  $\sim$  is the equivalence relation given by

$$x \sim y \iff \text{either } \begin{cases} x = y \\ \exists t \in [0, 1] \text{ such that } \{x, y\} = \{\gamma_{4i+1}(t), \gamma_{4i+3}(1-t)\} \text{ for some } i \\ \exists t \in [0, 1] \text{ such that } \{x, y\} = \{\gamma_{4i+2}(t), \gamma_{4i+4}(1-t)\} \text{ for some } i \end{cases}$$

The above construction is actually based on a simple idea. To build a sphere with  $g$  handles, we can just take  $g$  tori, make a cut in each, and glue them all together along these cuts. Any quadruple of edges  $\gamma_{4i+1}, \gamma_{4i+2}, \gamma_{4i+3}, \gamma_{4i+4}$  above forms one such torus. Very illustrative is the figure from [1].

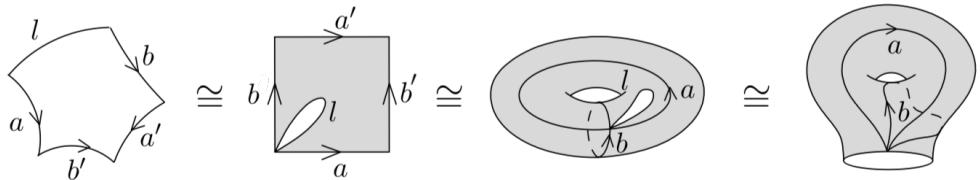


Figure 1.3: Torus being glued to the shape  $l$ .

We also note that the above construction can be described purely in topological terms. A sphere with  $g$  handles is (homeomorphic to) a connected sum  $\mathbb{T} \# \mathbb{T} \# \cdots \# \mathbb{T}$  of  $g$  tori  $\mathbb{T}$ .

## Riemann-Hurwitz

**Theorem 15** (Riemann-Hurwitz). *Let  $f : \mathcal{M} \rightarrow \mathcal{N}$  be a  $d$ -sheeted covering between compact Riemann surfaces. The genera  $g_{\mathcal{M}}, g_{\mathcal{N}}$  are then related by the following identity:*

$$g_{\mathcal{M}} = d(g_{\mathcal{N}} - 1) + 1 + \frac{b}{2}$$

where  $b$  is the total branching number defined as

$$b = \sum_{m \in \mathcal{M}} b_f(m)$$

In order to prove the Riemann-Hurwitz theorem, we need to introduce the notion of triangulation. A theorem due to Rado [13] states that every Riemann surface is triangulable.

**Definition 16** (Triangulation). A *triangulation*  $\tau$  of a compact topological surface  $\mathcal{S}$  will be understood to be the collection of sets  $V, E, F \subseteq \mathcal{S}$  where

- $V$  is a finite set of isolated points on  $\mathcal{S}$  - the *vertices*.
- $E = \{e_i\}$  is a finite set of the *edges* - the parametrised paths from  $e_i(0) \in V$  to  $e_i(1) \in V$ . Moreover, no two edges intersect elsewhere than vertices (that is,  $e_i((0, 1)) \cap e_j((0, 1)) = \emptyset$  for all  $i \neq j$ ).
- $F = \{f_i\}$  is the set of *faces* - the connected components of  $\mathcal{S} \setminus E$  with the requirement that for all closures  $\bar{f}_i$ , we have  $|\bar{f}_i \cap V| = 3$ . This condition gives the faces  $f_i$  a “triangle” shape.

**Definition 17** (Euler characteristic). The *Euler characteristic* of a compact topological surface  $\mathcal{S}$  is the quantity

$$\chi_{\mathcal{S}} := |V| - |E| + |F|$$

where  $(V, E, F)$  is some triangulation of  $\mathcal{S}$ .

For this to be well defined, the Euler characteristic must be invariant in the choice of triangulation of the surface, which turns out to be the case.  $\chi_{\mathcal{S}}$  is a topological (and homotopy) invariant.

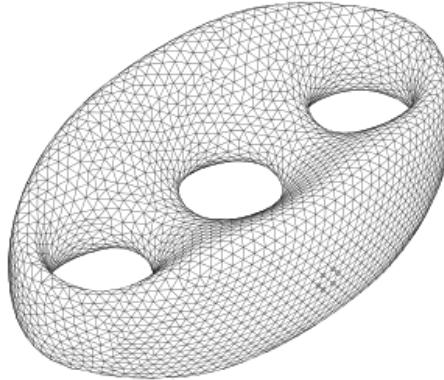


Figure 1.4: Triangulation of a surface of genus 3, Wikipedia [12].

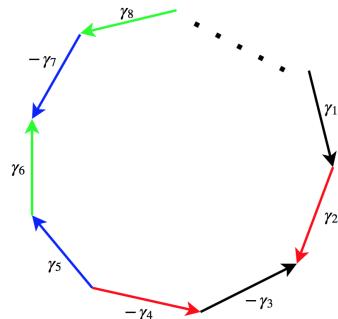


Figure 1.5: A sphere with handles.

**Theorem 18.** *The Euler characteristic  $\chi_{\mathcal{M}}$  of a compact Riemann surface  $\mathcal{M}$  is*

$$\chi_{\mathcal{M}} = 2 - 2g_{\mathcal{M}}$$

*Proof.* We start with the case of genus 0. The (unique) surface of genus 0 is the 2-sphere  $S^2$ . Considering a tetrahedron to be a triangulation of the sphere, we can easily compute

$$\chi_{S^2} = 4 - 6 + 4 = 2 = 2 - 2g_{S^2}$$

Consider now the case of genus 1. The (unique) surface of genus 1 is the 2-torus  $\mathbb{T}$ . Construct some triangulation  $\tau$  of the sphere which has two faces which do not share an edge or vertex. By cutting out these faces and identifying their boundary edges and vertices with the correct orientation (see figure), we obtain both the torus  $\mathbb{T}$  and its triangulation  $\tau'$ . Moreover, we have  $|V_{\tau'}| = |V_{\tau}| - 3$ ,  $|E_{\tau'}| = |E_{\tau}| - 3$  and  $|F_{\tau'}| = |F_{\tau}| - 2$ . This gives:

$$\chi_{\mathbb{T}} = |V_{\tau'}| - |E_{\tau'}| + |F_{\tau'}| = \chi_{S^2} - 2 = 0 = 2 - 2g_{\mathbb{T}}$$

We will finish the proof by showing that for any surface  $\mathcal{S}$  we have  $\chi_{\mathcal{S}\#T} = \chi_{\mathcal{S}} - 2$ . This means that gluing a torus to the surface decreases its Euler characteristic by 2.

Let  $\tau_{\mathcal{S}}, \tau_{\mathbb{T}}$  be triangulations of the surface  $\mathcal{S}$  and the torus respectively. Pick one face in  $\tau_{\mathcal{S}}$ , one in  $\tau_{\mathbb{T}}$  and identify their boundary edges and vertices with the correct orientation (see figure). Now we have obtained both the connected sum  $\mathcal{S}\#\mathbb{T}$  and its triangulation  $\tau_{\mathcal{S}\#\mathbb{T}}$ . We can finally compute:

$$\begin{aligned} \chi_{\mathcal{S}\#\mathbb{T}} &= (|V_{\mathcal{S}}| + |V_{\mathbb{T}}| - 3) - (|E_{\mathcal{S}}| + |E_{\mathbb{T}}| - 3) + (|F_{\mathcal{S}}| + |F_{\mathbb{T}}| - 2) \\ &= \chi_{\mathcal{S}} + \chi_{\mathbb{T}} - 2 \\ &= \chi_{\mathcal{S}} - 2 \end{aligned}$$

□

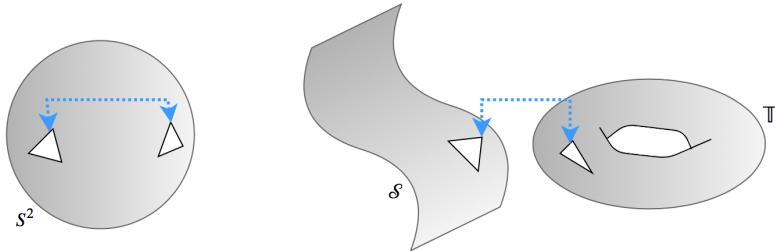


Figure 1.6: Change of Euler characteristic under “gluing”.

We are now in the position to prove the Riemann-Hurwitz theorem.

*Proof of the Riemann-Hurwitz theorem.* Given a  $d$ -sheeted covering  $f : \mathcal{M} \rightarrow \mathcal{N}$ , let  $B \subseteq \mathcal{M}$  be the set of branch points of the covering and consider a triangulation  $\tau = (V, E, F)$  of  $\mathcal{N}$  with  $f(B) \subseteq V$ . That is, every branch point is sent to a vertex in the triangulation.

But then, a triangulation  $\tau'$  is induced on  $\mathcal{M}$  by  $\tau' := f^{-1}(\tau) = (V', E', F')$ , where we can count the cardinalities:

$$|F'| = d|F|; \quad |E'| = d|E|; \quad |V'| = d|V| - b$$

with:

$$b = \sum_{m \in B} b_f(m) = \sum_{m \in M} b_f(m)$$

This relates the Euler characteristics  $\chi_M, \chi_N$  by:

$$\chi_M = d\chi_N - b$$

and we can finish off by using the relation  $\chi = 2 - 2g$ .

$$2 - 2g_M = d(2 - 2g_N) - b \implies g_M = d(g_N - 1) + 1 + \frac{b}{2}$$

□

## 1.5 Homology

We will now move on to study curves living on Riemann surfaces, as the last step before we are ready to treat integration. Recall the following definitions:

**Definition 19** (Path on surface). Let  $\mathcal{R}$  be a Riemann surface. We will understand a *path between  $r_1 \in \mathcal{R}$  and  $r_2 \in \mathcal{R}$*  to be a continuous map

$$\gamma_{r_1 r_2} : [0, 1] \rightarrow \mathcal{R}$$

**Definition 20** (Homotopy). Two paths  $\gamma_{xy}$  and  $\xi_{xy}$  are called *homotopic* if there exists a *homotopy*<sup>4</sup> between them. That is, a continuous function  $h : [0, 1] \times [0, 1] \rightarrow \mathcal{R}$  such that

$$h(-, 0) = \gamma_{xy} \text{ and } h(-, 1) = \xi_{xy}$$

Homotopy induces an equivalence relation  $\simeq$  whose equivalence classes are called *homotopic classes* and are denoted  $[\gamma_{xy}] = \{\xi_{xy} \mid \gamma_{xy} \simeq \xi_{xy}\}$ .

We can introduce a group structure on the set of homotopic classes - under the operation of concatenation. For two paths  $\gamma_{xy}, \xi_{yz}$  (notice that the final point and the initial point of the other one must coincide) the concatenation  $\gamma_{xy} \cdot \xi_{yz}$  to be a new path given by

$$\begin{aligned} \gamma_{xy} \cdot \xi_{yz} &: [0, 1] \rightarrow \mathcal{R} \\ \gamma_{xy} \cdot \xi_{yz} &= \begin{cases} \gamma_{xy}(2t) & \text{for } 0 \leq t \leq \frac{1}{2} \\ \xi_{yz}(2t) & \text{for } \frac{1}{2} \leq t \leq 1 \end{cases} \end{aligned}$$

The notion of “continuous deformation” given by homotopy is preserved by concatenation. (As can be verified) the concatenation is thus well-defined for the set of homotopic classes.

$$[\gamma] \cdot [\xi] = [\gamma \cdot \xi]$$

---

<sup>4</sup>To be more precise, a rel  $\{0, 1\}$  homotopy; one where the endpoints of paths are fixed.

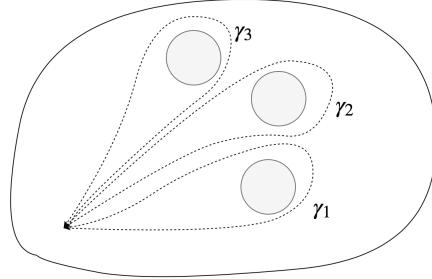
## Fundamental group

**Definition 21.** Let  $b$  be a point on a Riemann surface  $\mathcal{R}$ . The *fundamental group at point  $b$* , denoted  $\pi_1(\mathcal{R}, b)$ , is the set of homotopic classes  $[\gamma_{bb}]$  of *closed* paths  $\gamma_{bb} : [0, 1] \rightarrow \mathcal{R}$  with the operation of concatenation<sup>5</sup> defined above. Inverses in this group are given by path reversal (in the obvious sense) and the identity element is the homotopic class  $[e_b]$  of the constant path  $e_b : [0, 1] \rightarrow \mathcal{R}$  given by  $e_b(t) = b$ .

For path-connected  $\mathcal{R}$  (which is the case for Riemann surfaces in our setting) any two base points  $b$  give rise to isomorphic fundamental groups (the isomorphism is given by some choice of path between two distinct base points). This makes the base point argument in  $\pi_1(\mathcal{R}, b)$  redundant and we will from now on talk about the fundamental group as  $\pi_1(\mathcal{R})$ . Its identity will be referred to simply as 1.

Note that if, given a base point  $b$  on a surface of genus  $g$ , we encircle each of its  $g$  “holes” by paths  $\gamma_1, \gamma_2, \dots, \gamma_g$  (The fundamental group need not be commutative and these paths must be traversed in a specific order - see figure below.), the composition of these will be homotopy equivalent to a constant path. This implies that in the fundamental group we have the relation:

$$[\gamma_1] \cdot [\gamma_2] \cdot \dots \cdot [\gamma_g] = 1$$



Homotopy can be understood to capture information about holes on the surface. There are, however, more ways to capture this information - the concept of *homology* gives another approach, which will turn out to be more desirable for our purposes.

Consider any given triangulation  $\tau = (V, E, F)$  of a compact Riemann surface  $\mathcal{R}$ .

**Definition 22** ( $k$ -chains). We will call  *$0$ -chains* the formal sums<sup>6</sup> of vertices  $v_i \in V$ :

$$v = \sum_i n_i v_i$$

---

<sup>5</sup>The concatenation introduced above is not actually an associative relation;  $(\alpha \cdot \beta) \cdot \gamma$  is traversed under different speed than  $\alpha \cdot (\beta \cdot \gamma)$ . However, any two curves with different speeds arising from distinct bracketings of a sequence of concatenations will be homotopic to each other, hence the group operation is well-defined.

<sup>6</sup>Elements of the free abelian group with a given basis.

the 1-chains the formal sums of (oriented) edges  $e_i \in E$ :

$$e = \sum_i n_i e_i$$

and the 2-chains the formal sums of (oriented) faces/triangles  $f_i \in F$ :

$$f = \sum_i n_i f_i$$

$C_k$  will be the abelian group of  $k$ -chains, under addition. Note that all of the above sums are finite and coefficients  $n_i \in \mathbb{Z}$  are integers.

**Definition 23** (Boundary operator). We will also introduce the boundary operator  $\delta$ . Let  $[v_1, v_2]$  denote an oriented edge from vertex  $v_1$  to vertex  $v_2$ . The boundary operator is defined by:

$$\begin{aligned} \delta[v_1, v_2] &:= v_2 - v_1 \in C_0 && \text{for all } [v_1, v_2] \in E \\ \delta f &:= [v_1, v_2] + [v_2, v_3] + [v_3, v_1] \in C_1 && \text{for all } f \in F; \quad v_1, v_2, v_3 \text{ vertices of } f \end{aligned}$$

The boundary operator is naturally extended to the whole of  $C_1, C_2$  by linearity:

$$\begin{aligned} \delta \left( \sum_i n_i e_i \right) &= \sum_i n_i \delta e_i \\ \delta \left( \sum_i n_i f_i \right) &= \sum_i n_i \delta f_i \end{aligned}$$

and defines group homomorphisms  $\delta : C_1 \rightarrow C_0$  and  $\delta : C_2 \rightarrow C_1$ .

**Definition 24** (Cycles and boundaries). The subgroup  $C_\circ$  of  $C_1$  consists of *cycles*:

$$C_\circ := \{c \in C_1 \mid \delta c = 0\}$$

and the subgroup  $C_b$  of  $C_1$  consists of *boundaries*:

$$C_b := \delta C_2$$

**Theorem 25.** *For surfaces of genus greater than one,  $C_b < C_\circ < C_1$  (where  $<$  denotes the subgroup relation).*

*Proof.* We first remark that all three sets are groups:  $C_1$  by definition,  $C_b$  and  $C_\circ$  as a consequence of the  $\delta$  operator being linear. Thus we only need to show the proper containments.  $C_\circ \neq C_1$ , since any  $[v_1, v_2] \in C_1$  is not a cycle and  $C_\circ \subset C_1$  by definition. To see that  $C_b \neq C_\circ$  consider a cycle formed by edges going around the middle “hole” of a torus (or for higher genus surfaces, around any “hole”). This is not a boundary of any formal sum of faces (since removal of any such boundary would disconnect the torus into more connected components - which is not the case here). On the other hand,  $C_b \subset C_\circ$ , since  $\delta C_b = \delta \delta C_2 = \{0\}$  and thus any boundary is a cycle.  $\square$

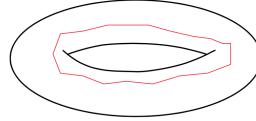


Figure 1.7: A cycle which is not a boundary.

So far, we have only taken into consideration the elements of  $C_o$  - the formal sums of edges in some triangulation  $\tau$ . We may, however extend the sets  $C_o$  and  $C_b$  into corresponding sets  $C'_o, C'_b$  containing all closed paths on the surface. This is due to the fact that any such closed path may be approximated by an element of  $C_o$  when a sufficiently fine triangulation is chosen - these two will be moreover homotopic<sup>7</sup>. In the opposite direction, any element of  $C_o$  already forms a closed path on the surface. Finally, it can also be shown [1] that the construction that we presented is independent of triangulation  $\tau$  considered.

**Definition 26** ((First) homology group). The first homology group of  $\mathcal{R}$  is the quotient

$$H_1(\mathcal{R}, \mathbb{Z}) := C'_o / C'_b$$

Two cycles  $c_1, c_2$  are *homologous* if they belong to that same coset. That is, if  $c_1 - c_2 \simeq \delta f$  for some  $f \in C'_2$ .

It can be shown (but we will not elaborate on this) that the first homology group is a finitely generated abelian group and that it is tied to the fundamental group by relation

$$H_1(\mathcal{R}, \mathbb{Z}) = \pi_1(\mathcal{R}) / [\pi_1(\mathcal{R}), \pi_1(\mathcal{R})]$$

**Definition 27** (Intersection number). For any two cycles  $\alpha, \beta$  with unit tangents  $t_\alpha, t_\beta$  that intersect in a discrete set of points, and any point  $P \in \mathcal{R}$  we define:

$$\alpha \circ \beta|_P := \begin{cases} +1 & \text{if } P \in \alpha \cap \beta \text{ and (vector product) } t_\alpha \times t_\beta \text{ points } \textit{out} \text{ of the surface} \\ -1 & \text{if } P \in \alpha \cap \beta \text{ and } t_\alpha \times t_\beta \text{ points } \textit{into} \text{ the surface} \\ 0 & \text{otherwise} \end{cases}$$

The *intersection number* of  $\alpha$  and  $\beta$  is

$$\alpha \circ \beta := \sum_{P \in \mathcal{R}} \alpha \circ \beta|_P$$

It can be seen that the intersection number is well-defined for the classes in  $H_1(\mathcal{R}, \mathbb{Z})$ . This is because the intersection number of any cycle with any boundary is 0 (the cycle enters and leaves the boundary equal number of times) and because we may represent any  $[\gamma], [\gamma'] \in H_1(\mathcal{R}, \mathbb{Z})$  by  $[\gamma] = \sum_i n_i \gamma_i$  and  $[\gamma'] = \sum_j m_j \gamma'_j$  so that they intersect in a discrete set of points. Moreover, we have

$$[\gamma] \circ [\gamma'] := \sum_{i,j} n_i m_j \gamma_i \circ \gamma'_j$$

---

<sup>7</sup>This can be constructed formally, see [11].

For clarity, we will now drop the class notation  $[-]$  for the elements of the homology group.

**Definition 28** (Canonical basis). A collection of elements  $a_1, a_2, \dots, a_g, b_1, b_2, \dots, b_g \in H_1(\mathcal{R}, \mathbb{Z})$  is a *canonical basis (of cycles)* of the first homology group of a Riemann surface  $\mathcal{R}$  if it generates the homology group and for the intersection numbers:

$$a_i \circ a_j = 0 \quad b_i \circ b_j = 0 \quad a_i \circ b_j = \delta_{ij} \quad \text{for all } i, j = 1, \dots, g$$

where  $\delta_{ij}$  is the Kronecker delta.

Any Riemann surface of genus  $g$  has a *non-unique* canonical basis. One choice for such a basis is, for example, depicted below.

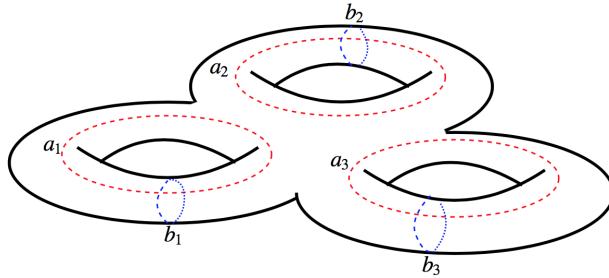


Figure 1.8: Canonical basis for a Riemann surface of genus 3.

To see the non-uniqueness part, we can see that other canonical bases may be generated by the elements of the symplectic group. Indeed, consider one given basis  $\mathbf{v} := (a_1 \ a_2 \ \dots \ a_g \ b_1 \ b_2 \ \dots \ b_g)^\top$ . Applying the intersection operation in the matrix form (in the obvious way), the constraint on the intersection numbers is precisely

$$\mathbf{v} \circ \mathbf{v}^\top = \begin{pmatrix} 0 & \mathbb{I}_g \\ -\mathbb{I}_g & 0 \end{pmatrix}$$

But then for any symplectic matrix  $S \in \mathrm{Sp}(2g, \mathbb{Z})$  we have by definition:

$$\begin{pmatrix} 0 & \mathbb{I}_g \\ -\mathbb{I}_g & 0 \end{pmatrix} = S^\top (\mathbf{v} \circ \mathbf{v}^\top) S = (S^\top \mathbf{v}) \circ (S^\top \mathbf{v})^\top$$

and thus  $S^\top \mathbf{v}$  is another canonical basis of cycles.

Finally, we remark that finding a canonical basis of cycles for a Riemann surface is not hard; the algorithmic method given by Tretkoff [5] which takes the monodromy representation of a surface as an input is manageable for hand calculations and works well on a computer. We will see later why it may be desirable to find specific canonical bases, constrained in some way - this will be precisely the purpose of the developed tool CyclePainter.

## 1.6 Monodromy

Before we continue with further theory related to homology, we introduce the monodromy representation of Riemann surfaces. This will be crucial in the algorithms introduced in the next chapter.

**Definition 29** (Monodromy group of a covering). Let  $f : \mathcal{M} \rightarrow \mathcal{N}$  be a  $d$ -sheeted holomorphic covering between Riemann surfaces and let  $p_m$  be a fixed base point on the surface  $\mathcal{N}$ . Any closed path  $\gamma$  on  $\mathcal{N}$  which starts and ends at  $p_m$  can be associated with an element  $\sigma_\gamma \in S_d$  of the symmetric group in the following way:

1. The lift  $f^{-1}(p_m) = (l_1, l_2, \dots, l_d)$  can be ordered.
2. As the parameter  $t \in [0, 1]$  varies from 0 to 1, the lift  $f^{-1}(\gamma(t))$  will vary continuously (by analytic continuation).
3. At  $t = 1$ , it will have arrived back to  $f^{-1}(p_m)$ , but now in the order  $(l_{\sigma_\gamma(1)}, l_{\sigma_\gamma(2)}, \dots, l_{\sigma_\gamma(d)})$ .

Given a specific ordering of  $f^{-1}(p_m)$ , we will call the *monodromy group of  $f$  at  $p_m$*  the subgroup of  $S_d$  generated by:

$$M(f, p_m) := \{\sigma_\gamma \mid \gamma \subset \mathcal{N} \text{ closed path starting and ending at } p_m\}$$

Note that different orderings of  $f^{-1}(p_m)$  give rise to different monodromy groups which are related together by conjugation.

Traversing the paths in reverse gives a relation worth remembering for the future use:  $\sigma_\gamma^{-1} = \sigma_{-\gamma}$ .

We will now scrutinize the connections between the monodromy group and the set of branch points of a covering. In the following discussion, consider the point  $p_m$  and initial ordering of  $f^{-1}(p_m)$  fixed. We include without a proof the (version of) central result of the monodromy theory - the Monodromy theorem.

**Theorem 30** (Monodromy theorem). *Let  $f : \mathcal{M} \rightarrow \mathcal{N}$  be a holomorphic covering with the set of branch points  $B$  and  $\alpha, \beta \subset \mathcal{N}$  two paths starting and ending at a fixed point  $p_m$ , which are homotopy equivalent in  $\mathcal{N} \setminus f(B)$ . Then  $\sigma_\alpha = \sigma_\beta$ .*

In the further text, we will synonymously use the word *monodromy* to refer to a set of generators of the monodromy group.

By the nature of holomorphic coverings (and bearing in mind Figure 1.1), it is easy to find one set of generators for the monodromy group. Consider the set  $f(B)$  of the images of branch points of  $f$ . For every  $b \in f(B)$ , we may define a path  $\gamma_b$  which starts at  $p_m$ , encircles  $b$  counterclockwise and comes back to  $p_m$  without having encircled any other element of  $f(B)$  (we will call  $\gamma_b$ 's the *monodromy paths*). Then, the set  $\{\gamma_b \mid b \in f(B)\}$  is a monodromy.

Indeed, the lift of any closed path  $\gamma$  that does not encircle any branch point will always give an identity permutation and thus  $f^{-1}(\gamma)$  consists of  $d$  disconnected elements. On the other hand, any non-identity permutation must have arisen from a path  $\gamma$  encircling some nonzero number of branch points. Such path will always be equivalent (using the relation  $\sigma_\gamma^{-1} = \sigma_{-\gamma}$ ) to a concatenation of (homotopic deformations) of  $\gamma_b$ 's. This technique will be used multiple times through the next chapter. For an example, see below.

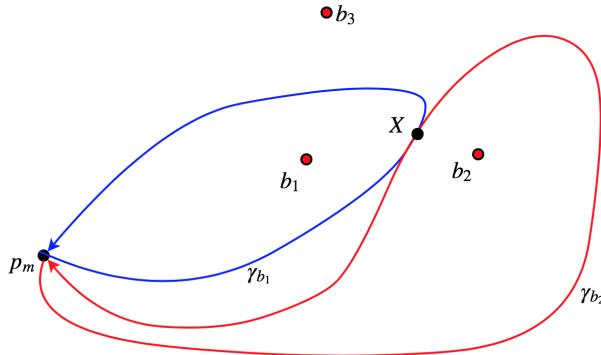


Figure 1.9: We can view the permutation<sup>9</sup>  $\sigma_{\gamma_{b_2} \cdot \gamma_{b_1}} = \sigma_{\gamma_{b_1}} \circ \sigma_{\gamma_{b_2}}$  given by concatenation in a different way - starting at  $p_m$ ; reach  $X$ ; go to  $p_m$  and back to  $X$  without encircling any branch point - this gives the identity permutation; continue from  $X$  to  $p_m$ . This will clearly give the same permutation. However, we may omit altogether the middle part, as its contribution is just the identity. Hence we are left only with encircling both  $b_1, b_2$  counterclockwise (A quicker argument is to see that  $\gamma_{b_2} \cdot \gamma_{b_1}$  is homotopy equivalent to encircling  $b_1, b_2$  counterclockwise)

We conclude the section by making a few important observations:

1. It is not difficult to see that choosing different monodromy paths will give rise to different sets of generators of the monodromy group. For two monodromy paths of the same branch point image, the Monodromy theorem will cease to hold if they are not homotopy equivalent in  $\mathcal{N} \setminus f(B)$ .
2. We may choose monodromy paths encircling the elements of  $f(B)$  arbitrarily closely - hence they give away information about the branch points of the covering. Specifically (recall Figure 1.1 again), for any  $b \in f(B)$  the number of disjoint cycles of length  $> 1$  in  $\sigma_{\gamma_b}$  gives the number of preimages  $f^{-1}(b)$ . The total number of branch points of the covering  $f$  is therefore:

$$|B| = \sum_{b \in f(B)} \sum_{k \in \mathbb{Z}_{\geq 2}} \# \text{ of } k\text{-cycles in } \sigma_{\gamma_b}$$

3. As a corollary, the permutations for any two monodromy paths corresponding to the same  $b \in f(B)$  have the same cycle type.

---

<sup>9</sup>In this context,  $\circ$  denotes function composition as usual. Traversing  $\gamma_{b_2}$  and then  $\gamma_{b_1}$  gives a permutation where first  $\sigma_{\gamma_{b_2}}$  is applied and then  $\sigma_{\gamma_{b_1}}$ . That is,  $\sigma_{\gamma_{b_1}} \circ \sigma_{\gamma_{b_2}}$ .

4. Any non-zero homology cycles will have to encircle branch points (otherwise they would be confined to the same sheet and homologous to 0). Thus, we can use monodromy information to study and reason about the homology. Specifically, in CyclePainter we will use it to scrutinize the image of the homology cycles under a holomorphic covering.

## 1.7 Holomorphic differentials & the period matrix

We now introduce the final algebraic tool needed for the applications.

**Definition 31** ( $k$ -forms). Let  $\mathcal{R}$  be a Riemann surface. We can assign to each local coordinate smooth complex valued functions  $f, p, q, s$  such that the following quantities are invariant (under the change of coordinate):

$$\begin{aligned} \text{0-forms/functions:} & \quad f = f(z, \bar{z}) \\ \text{1-forms/differentials:} & \quad \omega = p(z, \bar{z})dz + q(z, \bar{z})d\bar{z} \\ \text{2-forms:} & \quad S = s(z, \bar{z})dz \wedge d\bar{z} \end{aligned}$$

The set of  $k$ -forms is customarily denoted  $\Omega^k$ . Here  $\wedge$  denotes the standard exterior product and  $k$ -forms can be integrated over (finite unions of)  $k$ -chains. The exterior derivative operator  $d : \Omega^k \rightarrow \Omega^{k+1}$  sends  $k$ -forms to  $(k+1)$ -forms.

**Definition 32** (Exact/closed differentials). If  $\omega \in \Omega^1$  can be written as  $\omega = df$  for some  $f \in \Omega^0$ , it is *exact*. If  $d\omega = 0$ , it is *closed*.

The important theorem below connects closed differentials to the homology of the surface in a profound way.

**Theorem 33** (Stokes). *Let  $D$  be a 2-chain with a piecewise smooth boundary  $\partial D$  and  $\omega$  be a differential. The Stokes formula gives:*

$$\int_{\omega} d\omega = \oint_{\partial D} \omega$$

**Theorem 34.** *A differential  $\omega$  is closed if and only if for any two homologous paths  $\gamma_1, \gamma_2$  we have*

$$\int_{\gamma_1} \omega = \int_{\gamma_2} \omega$$

*Proof.* The forward implication is obvious. To see the backward implication, notice that a difference of any two homologous paths is by definition boundary  $\partial D$  of some domain  $D$ . We may thus employ the Stokes theorem:

$$\int_{\omega_1} - \int_{\omega_2} = \oint_{\partial D} \omega = \int_D d\omega = 0$$

□

Hence to be able to integrate closed differentials over arbitrary closed paths is the same as to be able to integrate closed differentials over the homology basis. This gives rise to the notion of periods.

**Definition 35** (Periods). Let  $\gamma_1, \gamma_2, \dots, \gamma_n$  be a collection of closed paths forming a homology basis and  $\omega$  be a closed differential. The periods of  $\omega$  are

$$\Lambda_i := \oint_{\gamma_i} \omega$$

This gives the following decomposition for an arbitrary closed curve  $\gamma = \sum_i n_i \gamma_i \in C_\circ$  on the surface:

$$\oint_{\gamma} \omega = \sum_i n_i \Lambda_i$$

If  $a_1, a_2, \dots, a_g, b_1, b_2, \dots, b_g$  is a canonical basis for the homology, we will denote the corresponding periods by  $A_i, B_i$ .

**Definition 36** (Holomorphic differential). A differential  $\omega$  is *holomorphic* if for any local coordinate it has the form  $\omega = \phi(z)dz$  with  $\phi$  holomorphic function. Clearly, holomorphic differentials are closed.

The set of all holomorphic differentials on the surface is denoted by  $H^1(\mathcal{R})$  and it is a  $\mathbb{C}$ -vector space. We state without proof that the dimension of  $H^1(\mathcal{R})$  is  $g_{\mathcal{R}}$  - the genus of the surface. We say that a set of  $g_{\mathcal{R}}$  linearly independent holomorphic differentials forms a basis for the *cohomology* of the surface.

We will refer to [3], listing further facts about the cohomology of an algebraic Riemann surface defined by a polynomial  $P(z, w)$  of total degree  $d$  (in  $z$  and  $w$ ):

1. The holomorphic differentials  $\omega_i$  in a cohomology basis are of the form

$$\omega_i = \frac{Q_i(z, w)}{\frac{\partial P}{\partial w}(z, w)} dz$$

where  $Q_i$  is a polynomial of degree (in  $z$  and  $w$ ) of degree at most  $d - 3$ .

2. If the Riemann surface is nonsingular, any collection of  $\frac{(d-1)(d-2)}{2}$  (this is the maximal number of such linearly independent polynomials) linearly independent polynomials  $Q_i$  of degree at most  $d - 3$  gives rise to a cohomology basis in the above form.
3.  $\implies$  the nonsingular algebraic Riemann surface in such a case has genus  $g_{\mathcal{R}} = \frac{(d-1)(d-2)}{2}$ .

**Definition 37** (Period matrix). Let  $\mathcal{R}$  be a Riemann surface of genus  $g$ ,  $a_1, a_2, \dots, a_g, b_1, b_2, \dots, b_g$  be a canonical basis for its homology and  $\omega_1, \omega_2, \dots, \omega_g$  be a basis for its cohomology. A *period matrix* of  $\mathcal{R}$  is the  $g \times 2g$  matrix given by:

$$\Omega_{\mathcal{R}} := (\mathbf{A} \quad \mathbf{B})$$

where

$$\begin{aligned}\mathbf{A} &= (A_{ij})_{i,j=1}^g & \mathbf{B} &= (B_{ij})_{i,j=1}^g \\ A_{ij} &= \oint_{a_j} \omega_i & B_{ij} &= \oint_{b_j} \omega_i\end{aligned}$$

The period matrix  $\Omega_{\mathcal{R}}$  depends on choices of both homology and cohomology bases. When different bases are chosen, different period matrices are obtained - up to this equivalence, however, the period matrix completely determines the Riemann surface. We may get rid of one of these dependencies by specifically choosing cohomology basis  $\{\tilde{\omega}_i\}$  for which

$$\oint_{a_i} \tilde{\omega}_i = \delta_{ij} \implies \mathbf{A} = \mathbb{I}_g$$

For such a choice of  $\{\tilde{\omega}_i\}$ ,  $\mathbf{B}$  does not depend on cohomology anymore - it only depends on the choice of homology basis. We will define this to be the Riemann matrix of  $\mathcal{R}$ . For any period matrix  $\Omega_{\mathcal{R}}$ , the matrix  $\mathbf{A}$  can be seen to be invertible.

**Definition 38** (Riemann matrix). Let  $\mathcal{R}$  be a Riemann surface with a period matrix  $\Omega_{\mathcal{R}} = (\mathbf{A} \ \mathbf{B})$ . The *Riemann matrix* of  $\mathcal{R}$  is the matrix given by:

$$\tau_{\mathcal{R}} := \mathbf{A}^{-1} \mathbf{B}$$

It can be shown that the Riemann matrix is always symmetric, with imaginary part positive definite.

## 1.8 Riemann surfaces with symmetries

We conclude the theory by a brief discussion about the symmetries that Riemann surfaces can possess. Riemann surfaces, describable as algebraic curves, naturally have automorphisms (and they form a group under composition). For example, any hyperelliptic curve will be sent to itself under

$$(x, y) \mapsto (x, -y)$$

Any Riemann surface with a large automorphism group will also have a homology (and cohomology) basis whose elements can be described as images of other elements - this will in turn constrain the Riemann matrix of the surface; there will very likely be a homology basis for which the Riemann matrix possesses a high degree of symmetry. Such results have been obtained for example by Braden in [4], utilizing the original CYCLEPAINTER for MAPLE. To present the functionality of CyclePainter, we will later take an example of the Klein's quartic from here. This is a Hurwitz curve which possesses the largest possible automorphism group for a surface of genus 3.

Perhaps surprisingly, the automorphism group of any compact Riemann surface of genus greater than 1 can be bounded.

**Theorem 39** (Hurwitz automorphism theorem). *Any compact Riemann surface  $\mathcal{R}$  of genus  $g_{\mathcal{R}} > 1$  has an automorphism group with order not exceeding*

$$84(g_{\mathcal{R}} - 1)$$

Any Riemann surface which achieves the Hurwitz bound is a *Hurwitz surface*. The Hurwitz bound is not achievable for any genera - it is achievable for example for  $g = 3$  (Klein's quartic) or  $g = 7$  (Macbeath's curve), but not any other  $1 < g < 7$ .

# Chapter 2

## CyclePainter

The main purpose of *CyclePainter* is to be a visual tool allowing an user to discover/build paths on a Riemann surface specified by its algebraic equation. The colour encoding of the surface’s sheets lets the user see the change of sheet along any path closed in  $x$ -projection, allowing him to readily identify closed cycles on the surface and process them further in computations.

The original *CyclePainter* program was written by Tim Northover [7] in 2009, using Maple and its *algcurves*. Throughout the years, it has become a legacy system and can no longer be run on new versions of Maple.

In this chapter, I will describe a Python port of this program into an open-source mathematics system Sage<sup>1</sup>, written for the purpose of this thesis, with some changes of functionality. This port, from now on, I will refer to simply as “CyclePainter”.

Throughout the chapter, let  $\mathcal{R}$  be a Riemann surface defined by algebraic equation  $f(x, y)$ , with the set of *finite* branch point  $x$ -projections  $\mathcal{B}$ . We will refer to these projections simply as “branch points” from now on. The case of  $\infty$  being a branch point will be covered separately where relevant.

### 2.1 Riemann surfaces in mathematics software systems

At the time of writing this thesis, the available tools for performing calculations on Riemann surfaces are present as packages within two popular mathematical pieces of software: Maple and Sage.

#### Maple - *algcurves*

The *algcurves* package built in Maple provides computational tools for studying 1-dimensional algebraic curves, defined by multivariate polynomials. It employs much of the Riemann surface theory covered in the introduction; to name a few the following methods are available: puiseux, monodromy, genus, differentials,

---

<sup>1</sup><http://www.sagemath.org>

singularities, periodmatrix, homology, ... A full documentation and description can be found at [10].

Despite the package having been a part of Maple for several years now, and working very reliably, for some applications algcurves lacks in flexibility. The monodromy, homology, differentials and thus the period matrix are all determined algorithmically and the user hence cannot specify their own homology or differentials for performing the calculations.

## Sage - abelfunctions

The *abelfunctions* module, written in Python by Chris Swierczewski [2], for the open-source mathematical system SageMath is an attempt to port the functionality of algcurves into Sage. In terms of functionality, abelfunctions provides mostly the same range of functions as algcurves, often making use of the same algorithms. There is currently work in progress to implement methods in which user will be able to provide custom homology basis and differentials. CyclePainter is meant to be complementary to this effort.

## 2.2 Crucial algorithms available

The abelfunctions module comes equipped with a collection of algorithms of which we will make extensive use in the implementation of CyclePainter. Below are listed the most important of them.

### Analytic continuation

Consider function  $c : \mathbb{C} \rightarrow \mathbb{C}^n$  (where  $n$  is the sheet number of  $\mathcal{R}$ ) defined implicitly by  $f(x, y) = 0$  sending  $x$  to fibre  $\mathbf{y} = \{y_1, y_2, \dots, y_n\}$ . This function is a continuous function of  $x$  and a big part of abelfunctions and CyclePainter relies on the ability to analytically continue  $c$  along paths. Note that at each point of the path, we can easily determine the fibre, but it is not straightforward to determine the order of its elements.

Given a path  $\gamma \subset \mathbb{C}$ , this is achieved in abelfunctions as follows:

- Pick a base point  $x_0$  and impose an ordering on  $c(x_0) = (y_1^{(0)}, y_2^{(0)}, \dots, y_n^{(0)})$ .
- Assume now that we have analytically continued up to point  $x_i$  on the path, resulting in ordering  $c(x_i) = (y_1^{(i)}, y_2^{(i)}, \dots, y_n^{(i)})$ .
- Pick a step size  $\delta$ , and choose a next point  $x_{i+1}$  on  $\gamma$  “ $\delta$  away” from  $x_i$ . Try the following:
- Build an approximate ordered fibre  $\mathbf{y}^\sim$  of roots at  $x_{i+1}$  by Newton’s approximation:

$$y_k^\sim = y_k^{(i)} - \delta \frac{f(x_i, y_k^{(i)})}{\frac{\partial f}{\partial y}(x_i, y_k^{(i)})}$$

- If  $\|\mathbf{y}^\sim - c(x_{i+1})\|$  is sufficiently small, match the approximate entries in  $\mathbf{y}^\sim$  by the true roots in  $c(x_{i+1})$ . Preserve this ordering and continue the analytic continuation for next  $x$ .
- If it is not sufficiently small, decrease  $\delta$  and repeat the above steps.

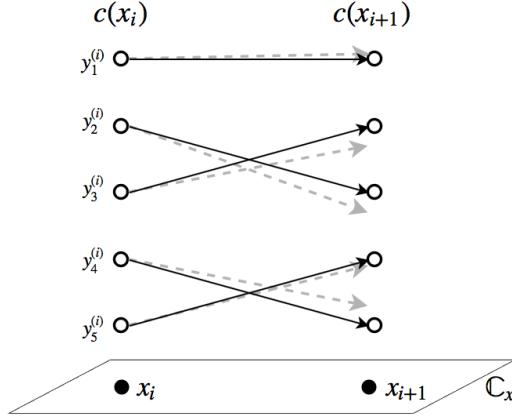


Figure 2.1: The dashed grey arrows indicate the Newton's approximation  $\mathbf{y}^\sim$ . The full black arrows show the actual matching of the roots in the continuation.

## Monodromy

A sketch of the algorithm for calculating monodromy in abelfunctions is the following:

- Pick a base point  $p_m$  and impose an ordering on  $c(p_m)$  as described in analytic continuation.
- Impose a particular ordering on the branch points.
- Construct a collection of monodromy paths around each branch point, in the above order.
- For each monodromy path, analytically continue the base fibre along each path.
- Upon returning to the base point, extract the monodromy permutation from how the fibre elements changes.

The nature of monodromy paths and further details will be mentioned in the section *Modifying the monodromy*.

## Homology and intersection numbers

Given the information about the monodromy group, an algorithm due to Tretkoff and Tretkoff [5] is used to find  $a$  and  $b$  cycles forming a canonical homology basis. The algorithm starts by choosing a base point  $x$  and constructing a certain non-directed graph with vertices at the elements of the fibre  $c(x)$  and all disjoint cycles present in the monodromy permutations of branch points  $b \in \mathcal{B}$ . Next, the graph is reduced to a spanning tree, and each edge that was removed in the process gives rise to a unique closed path on the Riemann surface (and unique cycle in the graph). An orientation is next chosen for such a path. Tretkoff and Tretkoff furthermore proved that the cycle constructed in this way are not contractible to a point and that the set of such cycles contains a basis for the homology. Finally, [5] also gives an graph-algorithmic way to find the intersection numbers of the cycles. This information is algebraically used to find linear combinations of these intermediate cycles that form the canonical basis of  $a$  and  $b$  cycles.

## Resolving singularities

Abelfunctions also implement the Puiseux series - these give directly a smooth parametrization around the singular points of a curve, allowing us effectively to handle singular algebraic Riemann surfaces.

## 2.3 Functionality

CyclePainter was developed as a tool to manipulate and explore paths on (compact/algebraic) Riemann surfaces. It is meant to be a handy and simple visual tool that does the necessary job<sup>2</sup>. The main use cases are *finding closed paths on the surface* (which in turn allows the user to *define a custom homology basis*) and *performing calculations on manually entered paths*. A creative user may however find other uses as well. Throughout the writing of this project, CyclePainter was useful for scrutinizing the monodromy group of a surface, identifying symmetries and mapping paths via automorphisms.

## How it works

Consider the holomorphic covering  $\pi_x : \mathcal{R} \rightarrow \mathbb{P}^1$  from  $\mathcal{R}$  to the Riemann sphere given by  $x$ -projection. Based on the  $x$ -projections  $\mathcal{B}$  of the branch points of this covering, we will first choose a set  $C \subset \mathbb{P}^1$  of *cuts* (note that the terminology “cut” is used for convenience and does not have any connection whatsoever to the concept of branch cuts from complex analysis). This set of cuts will be a star-shaped collection of edges between a chosen point  $p_c \in \mathbb{C} \setminus \mathcal{B}$  and every  $b \in \mathcal{B}$  (including infinity), such that no two branch point  $x$ -projections will end up on the same edge. Furthermore, we distinguish another point  $p_m \in \mathbb{C} \setminus \mathcal{B}$ .

---

<sup>2</sup>Conforming to the paradigm of any mathematical software.

If  $\pi_x$  is a  $d$ -sheeted covering, the set  $\pi_x^{-1}(\mathbb{P}^1 \setminus C) =: S = \{s_1, s_2, \dots, s_d\}$  comprises of precisely  $d$  disjoint connected components  $s_i$  (the *sheets*).

Next, we choose a bijective colour encoding  $\nu : \pi_x^{-1}(p_m) \rightarrow \text{Clr}$  which maps the elements of the fibre  $\pi_x^{-1}(p_m)$  to  $d$  distinct colours. This we extend to:

$$\begin{aligned}\nu : S &\rightarrow \text{Clr} \\ s_i &\mapsto \nu(s_i \cap \pi_x^{-1}(p_m))\end{aligned}$$

Note that colouring of the sheets also gives an *ordering* of the sheets. Without loss of generality, let this be from now on  $s_i \mapsto i$ . In particular, this induces an ordering on fibres  $\mathbf{y} = (y_1, \dots, y_d) := \pi_x^{-1}(x)$  for every  $x \in \mathbb{P}^1 \setminus C$ , defined by relation  $y_i \in s_i$ . We will consider the colour of a sheet and its index in ordering to be synonymous.

**Definition 40** (Visual space). The visual space of CyclePainter (i.e. what the user sees and on which he will draw paths) is the set

$$\{(\pi_x(s_i) \setminus \{\infty\}, \nu(s_i)) \mid i = 1, \dots, d\}$$

of  $d$  sheets'  $x$ -projections, each having a different fixed colour.

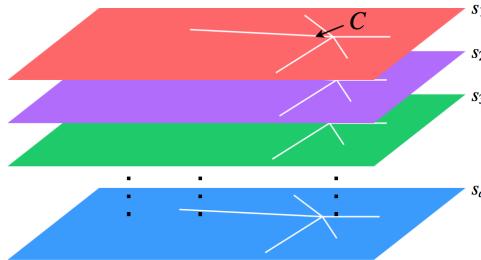


Figure 2.2: The visual space of CyclePainter.

Finally, we need to glue together these sheets of the Visual space along the cuts  $C$  in such a way that the *key property* for the functionality of CyclePainter will hold:

Consider any path  $\gamma$  in the Visual space starting on sheet  $s_i$  with  $x = p_m$  and ending on sheet  $s_j$  with  $x = p_m$ , such that its every intersection with  $C \setminus \{p_c\}$  is discrete. At each such intersection, we will (possibly) “jump” to a different sheet according to the gluing. We leave the behaviour along the cuts undefined.

Then if we analytically continued  $(p_m, (\pi_x^{-1}(p_m))_i)$  along the  $x$ -coordinates of  $\gamma$  on the original Riemann surface  $\mathcal{R}$ , we would end up at  $(\pi_x^{-1}(p_m))_j$ .

In words, we need to assure that moving between two  $y$ -solutions in the fibre of  $p_m$  along any path which starts and ends in this fibre is the same in the Visual space as it is in  $\mathcal{R}$ .

Note that finding a “gluing” is equivalent to defining a function for the change of colour when crossing any edge in  $C \setminus \{p_c\}$ . It turns out that there is always a specific monodromy on the surface which associates each  $b \in \mathcal{B}$  (and infinity) with a corresponding permutation  $\sigma(b)$ , which gives what we need. The construction of such monodromy is described later in the section [Modifying the monodromy](#). The function for the change of colour/sheet is then:

If we are on  $s_i$  and cross the edge  $[p_c, b]$   $\begin{cases} \text{counterclockwise:} & \text{change to } s_{\sigma(b)(i)} \\ \text{clockwise:} & \text{change to } s_{\sigma^{-1}(b)(i)} \end{cases}$

**Warning.** In practice, we will allow the user to start and end their path  $\gamma$  on any  $x$ -coordinate  $x_0 \notin C$ . But as the ordering was, in the first place, imposed at  $p_m$ , this will conceptually correspond to working with the path  $\gamma' + \gamma - \gamma'$  where  $\gamma'$  is some path between  $p_m$  and  $x_0$  which does not intersect any edge in  $C$  (and hence doesn’t encircle any branch point on  $\mathcal{R}$ ). This means two things:

- If  $\gamma$  is a closed path on  $\mathcal{R}$  (i.e. starts and ends on the same sheet), it is easy to see that so will be  $\gamma' + \gamma - \gamma'$  and vice versa. Hence the user will still detect closed paths.
- Starting on a sheet  $s_i$  at a point with  $x_0 \neq p_m$  means starting on the element of the fibre  $\pi_x^{-1}(x_0)$  which is reached by analytically continuing  $\pi_x^{-1}(p_m)_i$  along  $\gamma'$ .

For CyclePainter to function correctly, it is recommended that the user starts their path near the base point  $p_m$ . This is due to currently implemented representation of paths.

## User interface

CyclePainter is meant to be run within the JUPYTER notebook with installed SAGE and abelfunctions. The user will be able to interact with the program at the following places:

1. The title, showing the polynomial of a surface being examined and the number of saved paths.
2. The main clickable canvas, showing the branch points, cuts, point  $p_m$  and a path currently being built.
3. The sheet information sidebars - one for tying the sheets with their colours, the second one for changing the initial sheet of the path currently being built.
4. A collection of buttons for building paths and handling saved paths.
5. Calling methods in PYTHON of the CYCLEPAINTER object.

## Methods available to the user

Apart from the graphical interface for defining paths, the following auxiliary methods have been implemented:

- **SAVE/LOAD PATHS:** to save current path or to load a collection of paths previously defined from a file. Located in the graphical interface.
- **SHOW BRANCH PERMUTATIONS:** to show the branch point locations together with their associated monodromy permutations.

```
cp.show_branch_permutations()
```

- **GET  $x, y$  COORDINATES:** given an user-defined path in the Visual space, obtains the actual  $x, y$  coordinates of the corresponding path on the surface.

```
cp.get_xy_coordinates(path_index)
```

- **APPLY AUTOMORPHISM:** applies a given automorphism on a user-defined path.

```
cp.plot_automorphism_on_path(automorphism, path_index)
```

- **SHOW START/END FIBRE:** shows the starting and ending fibres of a user-defined path.

```
cp.start_fibre(path_index)  
cp.end_fibre(path_index)
```

- **GET  $x/y$  AT PARAMETER  $t$ :** shows the  $x$  value or  $y$  fibre at parameter  $t \in [0, 1]$  on a user-defined path. Generalized SHOW START/END FIBRE.

```
cp.build_RiemannSurfacePath(path_index).get_x(t)  
cp.build_RiemannSurfacePath(path_index).get_y(t)
```

- **GET INTERSECTION NUMBER/MATRIX:** given two user-defined paths/arrays of paths, finds their intersection number/matrix.

```
cp.intersection_matrix(index_array_1, index_array_2)
```

- **INTEGRATE:** given a holomorphic differential, integrates it along any user-defined path. The differential must be of the form  $f(x, y)dx$  and the user only needs to pass  $f(x, y)$  to the function.

```
cp.integrate(differential, path_index)
```

- **FIND PERIOD MATRIX:** given a canonical homology basis of user-defined paths, as well as the cohomology basis of differentials, calculates the period matrix.

```
cp.period_matrix(a_index_array, b_index_array, differential_array)
```

## 2.4 Calculating the point $p_c$

For latter algorithms to function correctly, it is necessary to find a suitable point  $p_c$  (representing the centre of the *star* formed by the cuts) satisfying the following constraints:

1. For any  $b \in \mathcal{B}$ , the points  $p_c, b$  and  $p_m$  (defined later, satisfies  $p_m = \text{Im } p_c$ ) must not be collinear.
2. For any distinct  $b_i, b_j \in \mathcal{B}$ ,  $b_j$  does not lie on the line segment  $b_ip_c$ .
3. For the sake of practical usability, no angle  $\angle b_ip_c b_j$  should be too small.

The strategy for finding such a point in CyclePainter is to build a set  $S$  of reasonable candidate points, picking the most suitable one. The heuristic used for suitability is to maximize the minimal angle between any two relevant points and the chosen candidate. It can be seen that this approach deals with all three constraints simultaneously. The algorithm in more detail is:

- Let  $\mathcal{S}$  be a set containing the centroid  $\frac{\sum_{b \in \mathcal{B}} b}{|\mathcal{B}|}$  and a small amount of randomly chosen points in its vicinity.
- Find the corners of the smallest rectangle in the complex plane which contains all of the finite branch points:

$$c_1 := \min_{b \in \mathcal{B}} \operatorname{Re} b + i \min_{b \in \mathcal{B}} \operatorname{Im} b$$

$$c_2 := \max_{b \in \mathcal{B}} \operatorname{Re} b + i \max_{b \in \mathcal{B}} \operatorname{Im} b$$

- Given a specific positive integer  $f$  representing fineness of the grid, build the “grid” set

$$G := \left\{ c_1 + m \frac{\operatorname{Re}(c_2 - c_1)}{f} + n \frac{i \operatorname{Im}(c_2 - c_1)}{f} \mid m, n \in \{0, \dots, f\} \right\}$$

- Choose  $p_c$  to be:

$$p_c := \arg \max_{c \in \mathcal{S} \cup G} \left[ \min_{\substack{p, q \in \mathcal{B} \cup \{c-1, c+1\} \\ p \neq q}} |\angle pcq| \right]$$

Where in the above expression we have added complex numbers  $c-1, c+1$  into the set for angle calculation to account for the fact that the cuts to infinity will be represented as horizontal lines.

## 2.5 Modifying the monodromy

### Problem description

For illustration purposes, in the figures to follow, we will work with a particular Riemann surface corresponding to the curve  $-x^4 + y^6 + 5y^3 + 1$ .

First, the point  $p_c$ , (representing the centre of the *star* formed by the cuts) is chosen, such that no two branch points and  $p_c$  are collinear. Consider another point  $p_m$ , such that:

$$\operatorname{Re} p_m \ll \min_{b \in \mathcal{B}} \operatorname{Re} b \text{ and } \operatorname{Im} p_m = \operatorname{Im} p_c$$

We will use  $p_m$  to be the base point for algorithmic calculation of the monodromy group.  $p_m$  is in CyclePainter represented by the black circular point. Next, we will introduce the cuts between  $p_c$  and each branch point. The cut to the  $\infty$  branch point will be represented by an infinite<sup>3</sup> ray emitting from  $p_c$  in the direction of  $p_c + 1$ .

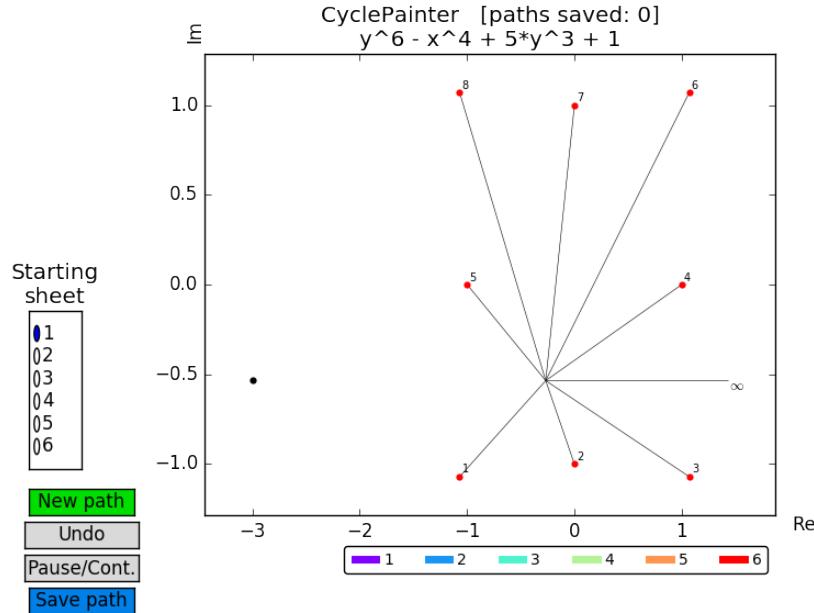


Figure 2.3: The set-up

Now, we will use *abelfunctions* to calculate the monodromy group. This will associate each branch point  $b \in \mathcal{B}$  with the permutation of sheets  $\pi(b)$ . The routine in *abelfunction* will have chosen the monodromy paths in the following manner:

- Consider a line segment  $l$  between  $p_m$  and the target branch point  $b_i$ .

---

<sup>3</sup>Although in CyclePainter this ray is illustrated finitely, it is effectively infinite.

- If for some  $j \neq i$   $b_j \in l$ , the path will avoid the point *clockwise* by a small half-circle.
- The point  $b_i$  is, at the end of the path, encircled *countrerclockwise*.
- The path returns to  $p_m$  along the same path.

In practice, these lines look as follows.

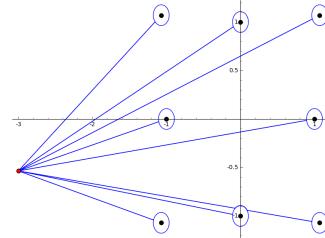


Figure 2.4: *abelfunctions'* monodromy paths

## Finding $\sigma$

To determine the colour changes at the cuts of CyclePainter, we will however need a monodromy with respects to paths where each path around branch point  $b_i$  intersects *only* the cut associated with  $b_i$  and *no other*. An example of such a path is below. Let  $\sigma(b)$  denote the permutation obtained after encircling branch point  $b$  along such a path. We will now describe an algorithmic way to obtain the permutations  $\sigma$  from the permutations  $\pi$  in time complexity  $\mathcal{O}(|\mathcal{B}|^2)$ .

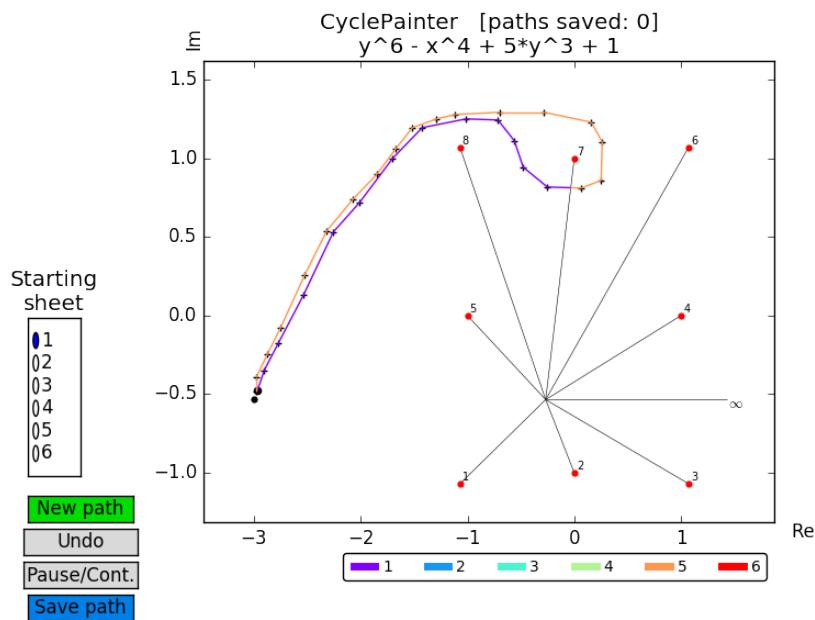


Figure 2.5: Path to a branch point avoiding cuts different from the cut of this particular branch point

Divide the branch points into two sets defined as follows<sup>4</sup>:

$$\mathcal{B}_L := \{b \in \mathcal{B} \mid \operatorname{Im} b > \operatorname{Im} p_c\}, \quad \mathcal{B}_R := \{b \in \mathcal{B} \mid \operatorname{Im} b < \operatorname{Im} p_c\}$$

We will calculate  $\sigma$  for  $\mathcal{B}_L$ ,  $\mathcal{B}_R$  and  $\{\infty\}$  separately.

$\sigma(\mathcal{B}_L)$

WLOG, consider the branch points  $b_1, b_2, \dots, b_{|\mathcal{B}_L|} \in \mathcal{B}_L$  to be ordered increasingly by the angle  $\angle p_m p_c b_i$ . We will compute  $\sigma(b_1), \sigma(b_2), \dots$  in this order - by imposing the ordering we will be able to assume that at the point of calculating  $\sigma(b_i)$ , all of the  $\sigma(b_j); j < i$  have been calculated already.

We will now calculate  $\sigma(b_i)$  as follows:

- Find the set  $S_i := \{b_j \mid j < i, \text{line } p_m b_i \text{ intersects the cut of } b_j \text{ and } b_j \notin \text{this line}\}$
- Preserving the above ordering (by angles) of  $b$ 's within the set  $S_i$ , denote these as  $s_1, s_2, \dots, s_{|S_i|}$
- Consider the permutation  $\omega$  specified by encircling precisely the branch points  $S_i \cup \{b_i\}$  counterclockwise. We will express  $\omega$  in two ways.
- First:

$$\omega = \sigma(s_{|S_i|}) \circ \dots \circ \sigma(s_3) \circ \sigma(s_2) \circ \sigma(s_1) \circ \pi(b_i)$$

To see this; to obtain  $\omega$ , we may first encircle  $b_i$  counterclockwise along the path used by *abelfunctions*, then encircle the elements of  $S_i$  one by one, as in the figure. Note that by assumption, all of the permutations  $\sigma$  being used have already been calculated.

---

<sup>4</sup>Note that  $p_c$  was chosen in such a way that for any branch point  $b$  we have  $\operatorname{Im} b \neq \operatorname{Im} p_c$ .

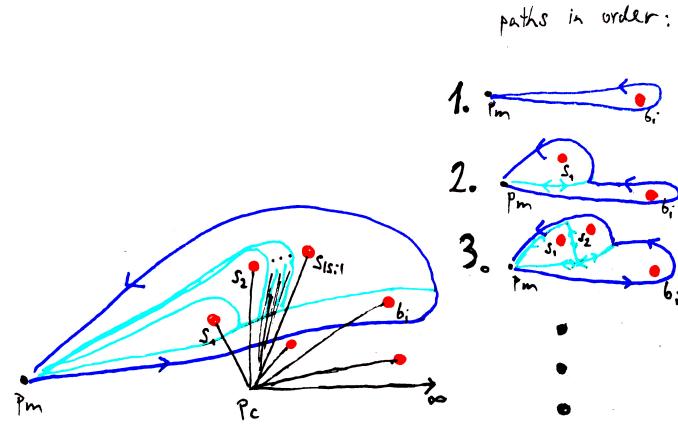
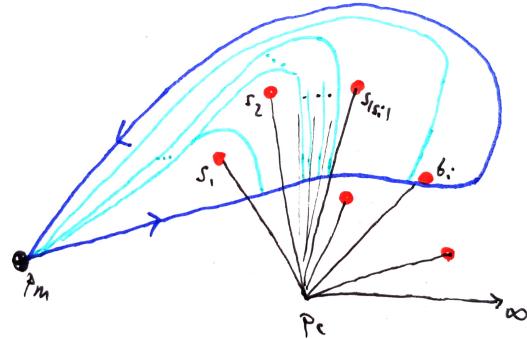


Figure 2.6: Cyan represents parts of path that have been cancelled by traversing them in both directions, blue represents the remaining parts.

- Secondly:

$$\omega = \sigma(b_i) \circ \sigma(s_{|S_i|}) \circ \cdots \circ \sigma(s_2) \circ \sigma(s_1)$$

Here, we will encircle counterclockwise the points in order  $s_1, s_2, \dots, s_{|S_i|}, b_i$ , as in the figure.



- From  $\sigma(s_{|S_i|}) \circ \cdots \circ \sigma(s_2) \circ \sigma(s_1) \circ \pi(b_i) = \sigma(b_i) \circ \sigma(s_{|S_i|}) \circ \cdots \circ \sigma(s_2) \circ \sigma(s_1)$  we obtain, via right multiplying the inverses:

$$\sigma(b_i) = \sigma(s_{|S_i|}) \circ \cdots \circ \sigma(s_3) \circ \sigma(s_2) \circ \sigma(s_1) \circ \pi(b_i) \circ \sigma^{-1}(s_1) \circ \sigma^{-1}(s_2) \circ \cdots \circ \sigma^{-1}(s_{|S_i|})$$

$$\sigma(\mathcal{B}_R)$$

In an analogous manner, we will construct  $\sigma$  for branch points in  $\mathcal{B}_R$ . Without loss of generality, consider  $b_1, b_2, \dots, b_{|\mathcal{B}_L|} \in \mathcal{B}_R$  to be ordered increasingly by the angle  $\angle p_m p_c b_i$ . Again, we will be computing  $\sigma$ 's in this order. To calculate  $\sigma(b_i)$ :

- Let  $S_i := \{b_j \mid j < i, \text{line } p_m b_i \text{ intersects the cut of } b_j \text{ or } b_j \in \text{this line}\}$

- Preserving the above ordering (by angles) of  $b$ 's within the set  $S_i$ , we have elements  $s_1, s_2, \dots, s_{|S_i|}$
- Consider the permutation  $\omega$  specified by encircling precisely the branch points  $S_i \cup \{b_i\}$  counterclockwise. We will express  $\omega$  in two ways.
- First:  

$$\omega = \pi(b_i) \circ \sigma(s_1) \circ \sigma(s_2) \circ \dots \circ \sigma(s_{|S_i|})$$
- Secondly:  

$$\omega = \sigma(s_1) \circ \sigma(s_2) \circ \dots \circ \sigma(s_{|S_i|}) \circ \sigma(b_i)$$

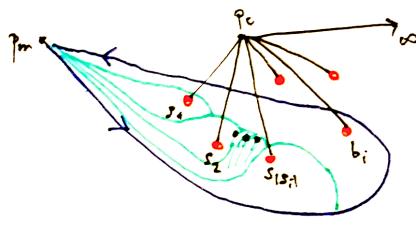


Figure 2.7: Using  $\pi(b_i)$

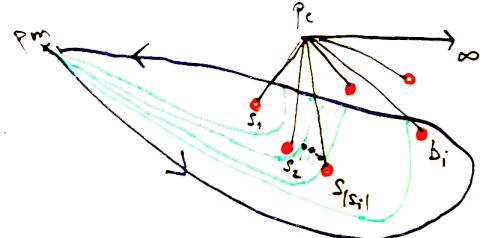


Figure 2.8: Using  $\sigma(b_i)$

- The above equations give us:

$$\sigma(b_i) = \sigma^{-1}(s_{|S_i|}) \circ \dots \circ \sigma^{-1}(s_2) \circ \sigma^{-1}(s_1) \circ \pi(b_i) \circ \sigma(s_1) \circ \sigma(s_2) \circ \dots \circ \sigma(s_{|S_i|})$$

$$\sigma(\infty)$$

To obtain  $\sigma(\infty)$ , we will use the fact that encircling all of the finite branch points clockwise is the same as encircling the infinity counterclockwise and vice versa.

Specifically, let  $b_1, b_2, \dots, b_{|\mathcal{B}|}$  be now the ordering of all of the finite branch points obtained by sweeping around  $p_c$ , starting from the infinity cut, clockwise. Then encircling all of the finite branch points counterclockwise gives the permutation

$$\sigma(b_{|\mathcal{B}|}) \circ \dots \circ \sigma(b_2) \circ \sigma(b_1)$$

and hence

$$\sigma(\infty) = \sigma^{-1}(b_1) \circ \sigma^{-1}(b_2) \circ \dots \circ \sigma^{-1}(b_{|\mathcal{B}|})$$

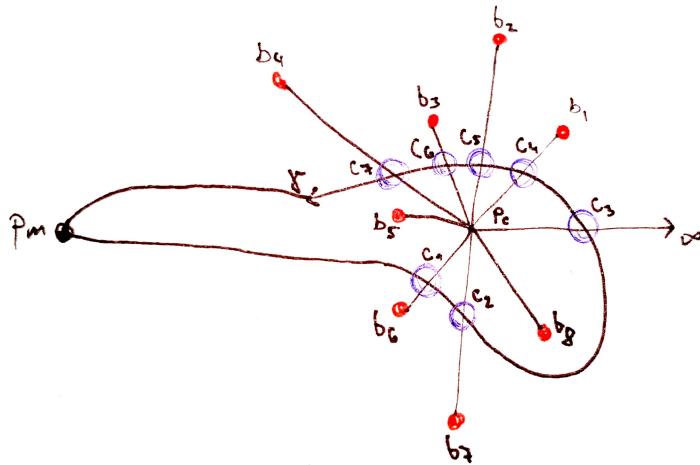
### (Further) justification for colours behaving as intended

Possessing now the monodromy  $\sigma$  constructed above, we will show that CyclePainter works as intended, according to the key property. It works as follows: the colour encoding of sheets directly follows  $\sigma$  - when a path crosses the cut of  $b \in \mathcal{B}$ , CyclePainter determines the new colour by applying  $\sigma(b)$  or  $\sigma^{-1}(b)$  depending whether the cut was crossed counterclockwise or clockwise.

**Proposition 41.** *The key property holds.*

Each change of colour along the path  $\gamma$  can be understood as encircling the corresponding branch point (in the corresponding direction) along its  $\sigma$ -monodromy path, back on  $\mathcal{R}$ . If the path obtained by such encirclings (this composition of encirclings is conceptual and has no connection to  $\gamma$ ) is equivalent to the path  $\gamma$ , we can assert that the overall change of colour really does correspond to the permutation of sheets that took place along  $\gamma$  on  $\mathcal{R}$ . We elaborate on this via diagrams in a specific example.

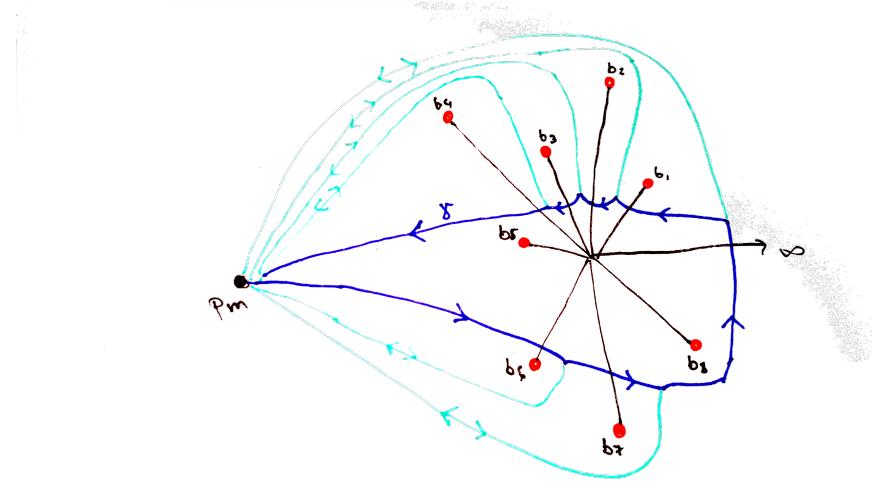
Below, we can see a path  $\gamma$  (the cases where  $\gamma$  does not cross the infinity cut can be handled analogously) with the places of change of colour denoted  $c_1, c_2, \dots$ .



For the depicted case, the overall change of colours will be given (as implemented in CyclePainter) by

$$\sigma(b_4) \circ \sigma(b_3) \circ \sigma(b_2) \circ \sigma(b_1) \circ \sigma^{-1}(\infty) \circ \sigma^{-1}(b_7) \circ \sigma^{-1}(b_6)$$

We can see below that the composition of the  $\sigma$ -monodromy paths corresponding to the above permutation is indeed equivalent to the path  $\gamma$ . As previously, cyan indicates the parts of the path cancelled by bidirectional traversal, while blue indicates the remaining parts of the path.



## Calculating intersection numbers

The above perspective of viewing drawn paths as equivalent to composition of  $\sigma$ -monodromy paths is useful for calculating the intersection number of two paths. The intersection count of two paths will be increased (or decreased, depending on the orientation) precisely when they intersect along any of their corresponding  $\sigma$ -monodromy paths. This is equivalent to two drawn paths intersecting while both *having the same colour*.

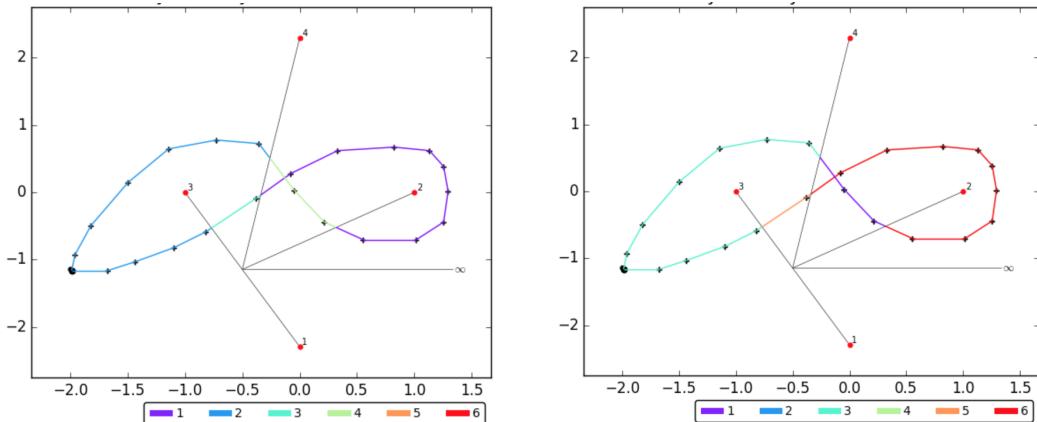


Figure 2.9: The intersection number of these two paths will be 1 - they intersect along the sheet 1 (purple).

## 2.6 Application of automorphisms

Let  $f : \mathcal{R} \rightarrow \mathcal{R}$  be an automorphism of the Riemann surface. The image of user-inputted paths under  $f$  is found in CyclePainter as follows:

- Analytically continue the whole  $y$  fibre along the user given path. From this, we will collect a list COOR of the  $x, y$  coordinates of the actual path on the surface.

- Apply pointwise  $f$  on the elements of  $\text{COOR}$ , obtaining a new list  $\text{COOR}_f$ .
- We may now plot the  $x$ -projection of  $\text{COOR}_f$  on the CyclePainter's canvas. The only remaining piece of information that we need is the starting sheet colour.

Currently, CyclePainter **does not** determine the correct starting sheet automatically. In the current version, the image will be correct only up to change of sheet - it is up to user to determine what the right starting sheet is. To do this, it is recommended to simply draw a path between  $p_m$  and its image  $\text{COOR}_f[0]_{(x)}$ , to see which is the sheet that reaches  $\text{COOR}_f[0]_{(y)}$ .

Moreover, it is recommended that the user redraws the whole image of the path so that it will start at  $p_m$ .

The whole procedure will be simplified in the later versions of CyclePainter.

## Source code of CyclePainter

The most recent version of the CyclePainter for abelfunctions is to be found here:

<https://github.com/markopoza/abelfunctions-cyclepainter>

# Chapter 3

## Examples

The below examples will also be available at the repository with the source code of CyclePainter.

### 3.1 A simple hyperelliptic curve

As a first worked example, we take a look at an hyperelliptic curve of genus 1, topologically a torus, given by:

$$P(x, y) = y^2 - (x^2 - 1)(x^2 - 4)$$

It is easy to see that the cycles encircling the middle two branch points and the two leftmost branch points should form a canonical homology basis. We will verify this.

1. Define the curve and the CyclePainter object.

```
> R.<x,y> = QQ[]
> cp = CyclePainter(y^2 - (x-1)*(x+1)*(x-2)*(x+2), monodromy_base_point=-5)
```

2. We examine the branch points. The monodromy is, unsurprisingly, the  $C_2$  group.

```
> cp.show_branch_permutations()
```

Branch point #1	2	(12)
Branch point #2	1	(12)
Branch point #3	-1	(12)
Branch point #4	-2	(12)

3. We draw and save the two described paths. They will be accessible in CyclePainter as paths with indices 0, 1 and we will refer to them as  $\mathfrak{a}$ ,  $\mathfrak{b}$  here. Their intersection matrix confirms that they are a canonical homology basis.

```

> cp.start()
> cp.intersection_matrix([0, 1], [0, 1])

```

$$\begin{pmatrix} \mathfrak{a} \\ \mathfrak{b} \end{pmatrix} \circ (\mathfrak{a} \quad \mathfrak{b}) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

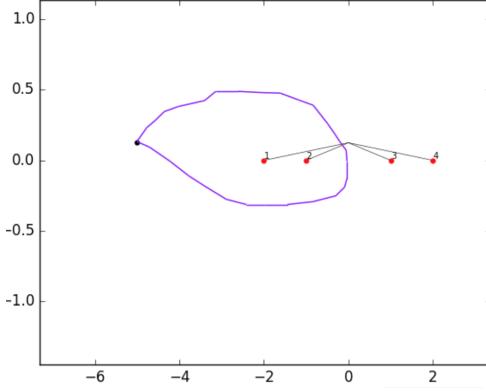


Figure 3.1:  $r\mathfrak{a}$

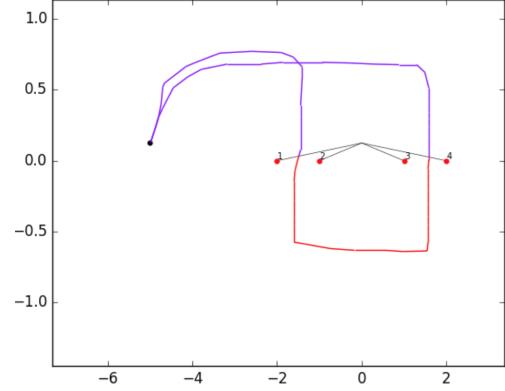


Figure 3.2:  $s\mathfrak{b}$

4. The only differential that we need is  $\omega = \frac{1}{y}$ . We can calculate the period and Riemann matrices with respect to our homology basis as follows:

```

> pm = cp.period_matrix([0], [1], [1/y])
> rm = pm[0, 1] / pm[0, 0]

```

$$\Omega = (-4.009.. - i0.069.. \quad 0.0512 + i0.000..) \quad \tau = (-0.003.. + i0.733)$$

## 3.2 Curve $x^2 = y^6 + 5y^3 + 1$

As a second worked example, we shall consider (the Riemann surface of) the curve  $\mathcal{R}$  given by  $P(x, y) = -x^2 + y^6 + 5y^3 + 1$  which has  $D_3$  symmetry given by automorphisms  $r, s : \mathcal{R} \rightarrow \mathcal{R}$ .

$$s(x, y) = (x, \rho y) \qquad \qquad r(x, y) = \left( \frac{x}{y^3}, \frac{1}{y} \right)$$

where  $\rho = e^{\frac{2\pi i}{3}}$  is the third root of unity.

The existence of a canonical homology basis of the form  $\{\mathfrak{a}, r\mathfrak{a}, s\mathfrak{a}, r(s\mathfrak{a})\}$  for a particular closed path  $\mathfrak{a}$  is shown in [8]. We shall use CyclePainter to construct such a homology basis and use it to calculate the Riemann matrix of the surface. The steps go:

1. We define the automorphisms.

```

> s = lambda x, y: (x, y*exp(2*pi*I/3).n())
> r = lambda x, y: (x/y^3, 1/y)

```

2. We define the curve and the CyclePainter object. The constant  $\kappa$  controls within ABELFUNCTIONS how close to branch points it will be possible to define paths. We have (on empirical grounds) chosen  $\kappa = \frac{1}{5}$ .

```
> R.<x,y> = QQ[]
> curve = -x^2 + y^6 + 5*y^3 + 1
> cp = CyclePainter(curve, kappa=1./5.)
> cp.start()
```

3. We examine the monodromy permutations at the branch points and their location:

```
> cp.show_branch_permutations()
```

#1	$-2.29128\dots i$	$(45)(36)(12)$
#2	1	$(164)$
#3	$-1$	$(235)$
#4	$2.29128\dots i$	$(56)(24)(13)$
#5	$+\infty$	$(253)(146)$

4. We draw the closed path  $\alpha$ , and show its image under automorphism  $s$ . Note that this is just a shift of sheets.

```
> cp.plot_automorphism_on_path(s, 0)
```

and save both. They will be accessible as paths with indices 0, 1.

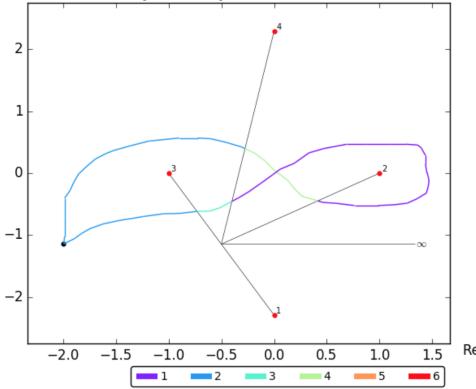


Figure 3.3:  $\alpha$

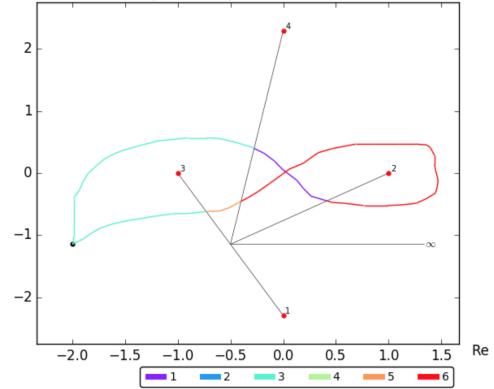


Figure 3.4:  $s\alpha$

5. We show the images of both  $\alpha$  and  $s\alpha$  under  $r$ .

```
> cp.plot_automorphism_on_path(r, 0)
> cp.plot_automorphism_on_path(r, 1)
```

As the point  $p_m$  will be sent to a different position under  $r$ , the result will be correct only up to a shift of sheets. We can however use the commands

```
cp.build_RiemannSurfacePath(i).get_x(t)
cp.build_RiemannSurfacePath(i).get_y(t)
```

to access the  $x$  position and ordered  $y$  fibre of any path with index  $i$ , at point corresponding to the parameter  $t \in [0, 1]$ . Hence we can figure out to which sheet the starting point of  $\alpha$  is sent under  $r$  (and analogously for the second path):

- We find that  $(p_m, y_k)$ , where  $\alpha$  starts at sheet  $k$ , is sent under  $r$  to  $(x', y')$ .
- We can construct manually a new “utility” path from  $p_m$  to  $x'$  which does not encircle any branch points.
- We examine the ending fibre of this utility path. If  $y'$  is in the fibre at position  $k'$ , our path  $r\alpha$  will need to start at sheet  $k'$ .

We find that  $r\alpha$  is supposed to start at sheet 3 and  $r(s\alpha)$  at sheet 2. We redraw and save both paths.

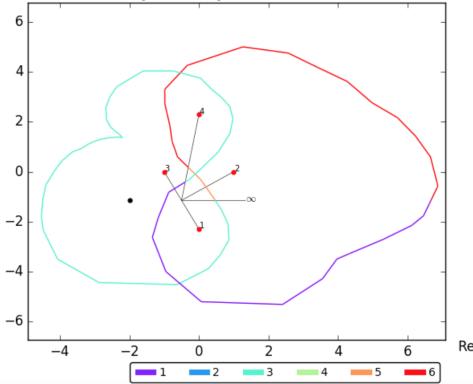


Figure 3.5:  $r\alpha$

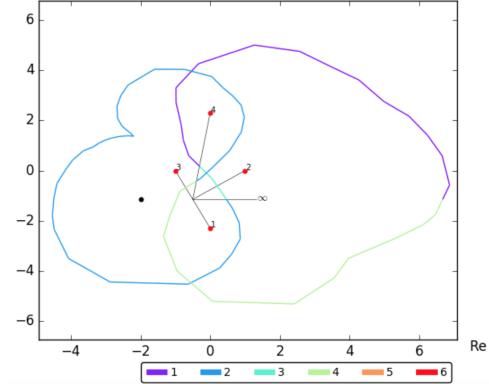


Figure 3.6:  $r(s\alpha)$

6. Now we verify that the cycles indeed form a canonical homology basis, using:

```
> a, sa, ra, rsa = 0, 1, 2, 3
> cp.intersection_matrix([a, ra], [sa, rsa])
```

we find that the intersection numbers of our paths are:

$$\begin{array}{lll} \alpha \circ s\alpha = 1 & \alpha \circ r\alpha = 0 & \alpha \circ r(s\alpha) = 0 \\ s\alpha \circ r\alpha = 0 & s\alpha \circ r(s\alpha) = 0 & r\alpha \circ r(s\alpha) = 1 \end{array}$$

Which means that this is indeed a canonical homology basis, considering  $\alpha, r\alpha$  to be the  $a$ -cycles and  $s\alpha, r(s\alpha)$  to be the  $b$ -cycles.

7. We will now let  $\omega_1 = \frac{1-y}{6y^5+15y^2}dx$  and  $\omega_2 = \frac{1+y}{6y^5+15y^2}dx$  to be the basis of holomorphic differentials. These have the nice property that  $r^*\omega_1 = \omega_1$  and  $r^*\omega_2 = -\omega_2$ . As shown in [8], this property should result in the following constraints on integrals over the cycles:

$$\begin{aligned}\oint_{\mathfrak{a}} \omega_1 &= \oint_{r\mathfrak{a}} \omega_1 & \oint_{s\mathfrak{a}} \omega_1 &= \oint_{r(s\mathfrak{a})} \omega_1 \\ \oint_{\mathfrak{a}} \omega_2 &= - \oint_{r\mathfrak{a}} \omega_2 & \oint_{s\mathfrak{a}} \omega_2 &= \oint_{r(s\mathfrak{a})} \omega_2\end{aligned}$$

```
> differentials = [(1-y)/(6*y^5 + 15*y^2), (1+y)/(6*y^5 + 15*y^2)]
```

8. By finding the period matrix, we verify that the above constraints indeed hold<sup>1</sup>.

```
> cp.period_matrix([a, ra], [sa, rsa], differentials)
```

$$\begin{pmatrix} -0.015.. - i0.004.. & -0.015.. - i0.004.. & 0.015.. - i0.004.. & 0.015 - i0.004.. \\ -0.007.. - i0.008.. & 0.007 + i0.008.. & 0.007.. - i0.008.. & -0.007.. + i0.008..\end{pmatrix}$$

### 3.3 Klein's curve

We shall now take a look at the Klein's quartic ( $\mathcal{K}$ ) - a compact Riemann surface of genus 3, which has the largest possible automorphism group. It is usually characterized as a projective algebraic curve with equation:

$$x^3y + y^3z + z^3x = 0$$

and the affine projection:

$$x^3y + y^3 + x = 0$$

It reaches the Hurwitz bound and its automorphism group is isomorphic to  $\text{PSL}(2, 7)$ , having 168 elements. In choosing the homology basis, we will trace the steps of [4], where Braden and Northover also show that via birational transformations, the curve may be re-expressed in different coordinates as:

$$w^7 = (z - 1)(z - \rho)^2(z - \rho^2)^4$$

where  $\rho = e^{\frac{2\pi}{3}}$  is the third root of unity. We first examine the curve in these coordinates. The curve has an order three automorphism given by  $s : \mathcal{K} \rightarrow \mathcal{K}$ :

$$s(z, w) = \left( \rho^2 z, \frac{\rho^2(z - 1)(z - \rho)(z - \rho^2)^2}{w^3} \right)$$

1. We set up the field that we will work in and define the Klein's quartic:

```
> k.<rho>=CyclotomicField(3)
> A2.<z,w>=AffineSpace(k,2)
> klein = w^7-(z-1)*(z-rho)^2*(z-rho^2)^4
```

---

<sup>1</sup>Up to small numerical errors.

2. We define the automorphism in code:

```
> s = lambda z, w: (rho^2*z, rho^2*(z-1)*(z-rho)*(z - rho^2)^2/w^3)
```

3. The covering will be 7-sheeted with the monodromy:

```
> cp = CyclePainter(klein)
> cp.show_branch_permutations()
```

#1	$-0.50000.. - i0.86602..$	(1725436)
#2	1	(1246753)
#3	$-0.50000.. + i0.86602..$	(1473265)

4. We define (and save) the closed path  $\alpha_1$  from which we will derive the other cycles in the homology basis. We also produce the image of  $\alpha_1$  under  $s$ ; this will be, again, correct only up to shift of sheets.

```
> cp.start() # now we draw a1
> cp.plot_automorphism_on_path(s, 0)
```

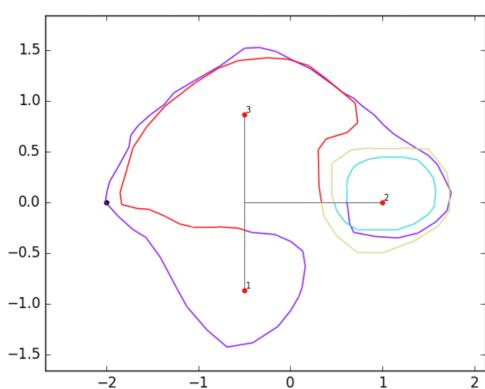


Figure 3.7:  $\alpha_1$

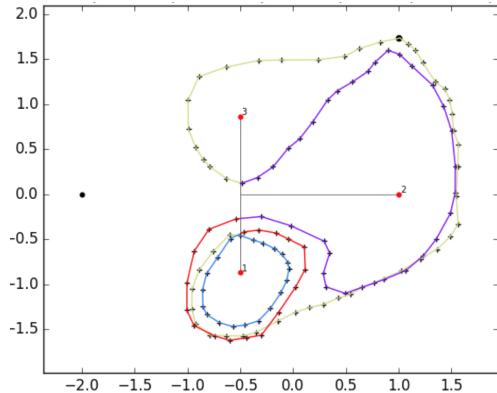


Figure 3.8: CyclePainter plot of  $s\alpha_1$ . We will need to shift sheets to be correct.

5. As previously, by noticing where the starting point of  $\alpha_1$  is sent, and examining the fibres, we find that  $s\alpha_1$  is supposed to start at sheet 1. As the image of  $p_m$  under  $s$  is also far from  $p_m$ , we will redraw this path manually so it starts close to  $p_m$  (for computational stability).

In this way we obtain  $\alpha_2 := s\alpha_1$ ; and completely analogously  $\alpha_3 := s(s\alpha_1)$ .

```
> cp.plot_automorphism_on_path(lambda x, y: s(*s(x, y), 0))
```

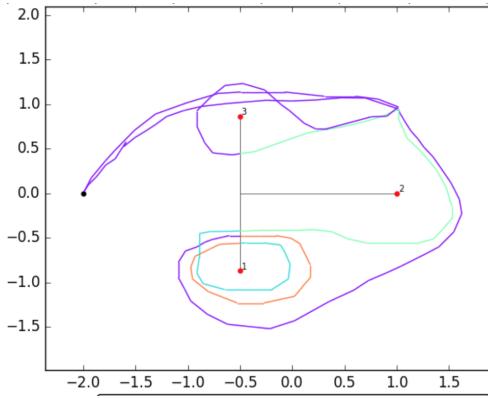


Figure 3.9: (Manually redrawn)  $\alpha_2$

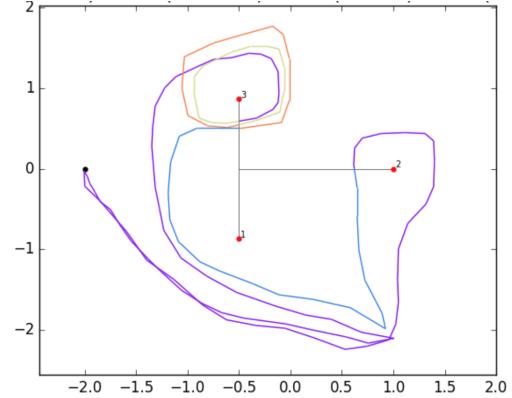


Figure 3.10: (Manually redrawn)  $\alpha_3$

6. By [4], there should be a collection of  $\mathfrak{b}$ -cycles obtained in the same way, by applying  $s$ . Here,  $\mathfrak{b}_1$  should be related to  $\alpha_1$  by a simple shift of sheets. By trying out all of the possible shifts of sheets of  $\alpha_1$ , we find that the one starting at sheet 4 is the right one:

```
> cp.intersection_matrix([0, 1, 2], [3])
```

$$\alpha_1 \circ \mathfrak{b}_1 = 1$$

$$\alpha_2 \circ \mathfrak{b}_1 = 0$$

$$\alpha_3 \circ \mathfrak{b}_1 = 0$$

We note that this was the only choice with the above intersection numbers.

7. We apply the automorphism to obtain all  $\mathfrak{b}$ -cycles.

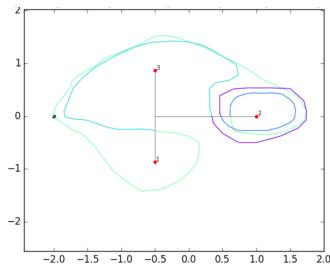


Figure 3.11:  $\mathfrak{b}_1$

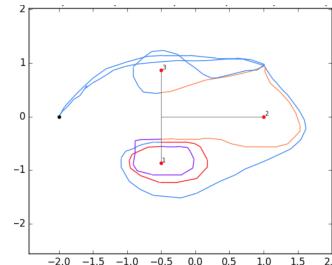


Figure 3.12:  $\mathfrak{b}_2$

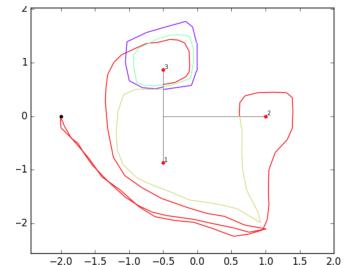


Figure 3.13:  $\mathfrak{b}_3$

8. This is indeed a canonical homology basis.

```
> cp.intersection_matrix(range(6), range(6))
```

$$\mathbf{v} := (\alpha_1 \ \alpha_2 \ \alpha_3 \ \mathfrak{b}_1 \ \mathfrak{b}_2 \ \mathfrak{b}_3)^T \implies \mathbf{v} \circ \mathbf{v}^T = \begin{pmatrix} 0 & \mathbb{I}_3 \\ -\mathbb{I}_3 & 0 \end{pmatrix}$$

9. As described in [4], there exists a cohomology basis which behaves well with respect to the symmetries of Klein's curve. This basis can also be acted on by the order 3 symmetry  $s$ , which only cyclically permutes the holomorphic differentials. In particular:

```

> differentials = [
>     (2*rho + 1)*(rho - z)*(rho^2 - z)^2/(7*w^5),
>     -(2*rho + 1)*(rho - z)*(rho^2 - z)^3*rho/(7*w^6),
>     (2*rho + 1)*(rho^2 - z)*rho^2/(7*w^3)
> ]

```

10. With these, we find the period matrix and the Riemann matrix:

```

> PM = cp.period_matrix([0,1,2],[3,4,5],differentials_revised)
> A, B = PM[:, :3], PM[:, 3:]
> RM = np.matmul(np.linalg.inv(A), B)
> for i in range(len(RM)):
>     print(' | '.join(map(str, RM[i, :])))

```

which gives (a very symmetric) Riemann matrix:

```

(-0.248.. + 0.663..j) | ( 0.432.. - 0.003..j) | ( 0.493.. + 0.001..j)
( 0.568.. - 0.001..j) | (-0.249.. + 0.656..j) | ( 0.563.. - 0.001..j)
( 0.513.. + 0.001..j) | ( 0.432.. + 0.005..j) | (-0.248.. + 0.659..j)

```

11. We notice that the above result is (up to inaccuracies originating in numerical integration) indeed consistent with the predicted theoretical form of the Riemann matrix which comes from [4] and is:

$$\tau = \frac{1}{2} \begin{pmatrix} e & 1 & 1 \\ 1 & e & 1 \\ 1 & 1 & e \end{pmatrix} \quad e = \frac{-1 + i\sqrt{7}}{2}$$

# Chapter 4

## Conclusion

Two main objectives of this project have been achieved; in the first chapter a self-contained undergraduate-level introduction to the theory of Riemann surfaces was given, deriving especially the aspects that are subject to computational treatment.

In the second chapter, we gave a brief description of the current state of affairs of available computational tools for Riemann surfaces. This was followed by an algorithmic and functional description of the tool CyclePainter, including general guidelines for its use.

As part of this project, the (first version of) CyclePainter for abelfunctions has been implemented. The third chapter consisted fully of worked use cases with this piece of software.

Admittedly, even though functional, CyclePainter in its current form has its deficiencies. I genuinely hope and plan to introduce improvements to the software, eventually with the prospect of the CyclePainter becoming a part of abelfunctions themselves.

The main goals for the future development of CyclePainter are:

- Resolve the numerical inaccuracies in integration.
- Get rid of the dependency of paths starting at a particular point.
- Optimize the performance.
- Refactor the source code for an easy integration with abelfunctions.

# Bibliography

- [1] Alexander Bobenko. *Compact Riemann Surfaces*. Lecture notes [Differential-geometrie III]. 2018  
<http://page.math.tu-berlin.de/~bobenko/Lehre/Skripte/RS.pdf> (2018)
- [2] C. Swierczewski and others. *Abelfunctions: A library for computing with Abelian functions, Riemann surfaces, and algebraic curves*.  
<http://github.com/abelfunctions/abelfunctions> (2017)
- [3] Bernard Deconinck and Mark Van Hoeij. *Computing Riemann matrices of algebraic curves*. Physica D vol. 152 (1999)
- [4] H. W. Braden and T.P. Northover. *Klein's Curve* J. Phys. A 43, 434009, [arXiv:0905.4202] (2010)
- [5] C. L. Tretkoff and M. D. Tretkoff. *Combinatorial group theory, Riemann surfaces and differential equations*. Contemporary Mathematics, 33:467-517 (1984)
- [6] C. Teleman. *Riemann Surfaces*. Lecture notes.  
<https://math.berkeley.edu/~teleman/math/Riemann.pdf> (2003)
- [7] T. P. Northover. *Riemann surfaces with symmetry: algorithms and applications*. Doctor of Philosophy thesis, University of Edinburgh. Edinburgh research archive.  
<http://hdl.handle.net/1842/9848> (2011)
- [8] H. W. Braden. *Riemann Surfaces with Symmetry*. Unpublished Notes, 26pp. (2006)
- [9] H. McKean and V. Moll. *Elliptic Curves: Function Theory, Geometry, Arithmetic*. ISBN 978-0521658171, Cambridge University Press; New Ed edition (13 Aug. 1999)
- [10] *Maplesoft - Overview of algcurves Package*.  
<https://www.maplesoft.com/support/help/Maple/view.aspx?path=algcurves> (accessed February 03, 2018)
- [11] Springer, G. *Introduction to Riemann Surfaces*. Chelsea Publishing Co., New York (1981)

- [12] *Surface triangulation*. On Wikipedia.  
[https://en.wikipedia.org/wiki/Surface\\_triangulation](https://en.wikipedia.org/wiki/Surface_triangulation)  
(accessed February 03, 2018)
- [13] Rado, Tibor. *Uber den Begriff der Riemannschen Flache*.  
Acta Szeged, 2 (2): 101-121 (1925)