2-group Belyi maps

Michael Musty

July 9, 2019

Dartmouth College

Outline

Motivation

Factoring in number fields

Let F be a number field with ring of integers $\mathbb{Z}_F$.

$$\begin{tikzcd}[column sep=small] & F \arrow[dl, dash] \arrow[d, dash] \\ \mathbb{Z}_F \arrow[d, dash] & \mathbb{Q} \arrow[dl, dash] \\ \mathbb{Z} & \end{tikzcd}$$

Factoring in number fields

Let F be a number field with ring of integers $\mathbb{Z}_F$.

$$\begin{tikzcd}
 & F \arrow[dl, dash] \arrow[d, dash] \\
\mathbb{Z}_F \arrow[d, dash] & \mathbb{Q} \arrow[dl, dash] \\
\mathbb{Z} &
\end{tikzcd}$$

$$p\mathbb{Z}_F = \mathfrak{p}_1^{e_1} \cdots \mathfrak{p}_g^{e_g}$$

Factoring in number fields

Let F be a number field with ring of integers $\mathbb{Z}_F$.

$$\begin{tikzcd} & F \\ \mathbb{Z}_F \arrow[ur, dash] & \mathbb{Q} \arrow[u, dash] \\ \mathbb{Z} \arrow[u, dash] \arrow[ur, dash] & \end{tikzcd}$$

$$p\mathbb{Z}_F = \mathfrak{p}_1^{e_1} \cdots \mathfrak{p}_g^{e_g}$$

A prime $p \in \mathbb{Z}$ is **ramified** in F if $e_i \geq 2$ for some i.

Factoring in number fields

Let F be a number field with ring of integers $\mathbb{Z}_F$.

$$\begin{tikzcd} & F \arrow[dl, dash] \arrow[d, dash] \\ \mathbb{Z}_F \arrow[d, dash] & \mathbb{Q} \arrow[dl, dash] \\ \mathbb{Z} & \end{tikzcd}$$

$$p\mathbb{Z}_F = \mathfrak{p}_1^{e_1} \cdots \mathfrak{p}_g^{e_g}$$

A prime $p \in \mathbb{Z}$ is **ramified** in F if $e_i \geq 2$ for some i.

Does there exist a number field where 2 is the *only* ramified prime?

Factoring in number fields

Let F be a number field with ring of integers $\mathbb{Z}_F$.

$$\begin{tikzcd} & F \arrow[dl, dash] \arrow[d, dash] \\ \mathbb{Z}_F \arrow[d, dash] & \mathbb{Q} \arrow[dl, dash] \\ \mathbb{Z} & \end{tikzcd}$$

$$p\mathbb{Z}_F = \mathfrak{p}_1^{e_1} \cdots \mathfrak{p}_g^{e_g}$$

A prime $p \in \mathbb{Z}$ is **ramified** in F if $e_i \geq 2$ for some i.

Does there exist a number field where 2 is the *only* ramified prime?
Nonsolvable?

Why 2-group Belyi maps?

$$
\begin{tikzcd}
\text{2-group Belyi maps} \arrow[rr, dashed] \arrow[dr, "\text{Beckmann's theorem}"] & & \text{number fields ramified at 2} \\
& \text{curves with good reduction away from } p = 2 \arrow[ur, "\text{Néron-Ogg-Shafaravich}"] &
\end{tikzcd}
$$

Why $p = 2$?

Conjecture (Gross 1998)

For every prime p, there exists a nonsolvable Galois number field ramified only at p.

Why $p = 2$?

Conjecture (Gross 1998)

For every prime p, there exists a nonsolvable Galois number field ramified only at p.

$p \geq 11$: existence (Serre), explicit (Edixhoven, Mascot)

$p = 7$: existence (Dieulefait, Roberts)

$p = 5$: existence (Dembélé, Greenberg, Voight), explicit (Roberts)

$p = 3$: existence (Dembélé, Greenberg, Voight)

$p = 2$: existence (Dembélé)

Why $p = 2$?

Conjecture (Gross 1998)

For every prime p, there exists a nonsolvable Galois number field ramified only at p.

$p \geq 11$: existence (Serre), explicit (Edixhoven, Mascot)

$p = 7$: existence (Dieulefait, Roberts)

$p = 5$: existence (Dembélé, Greenberg, Voight), explicit (Roberts)

$p = 3$: existence (Dembélé, Greenberg, Voight)

$p = 2$: existence (Dembélé)

The hope is that an explicit nonsolvable field ramified only at 2 can be obtained as $K(\mathrm{Jac}(X)[2])$ where X is the domain of a 2-**group Belyi map** (which we will define shortly).

Main results

Motivated by the applications of 2-group Belyi maps to arithmetic geometry, we now state the main results.

Main results

Motivated by the applications of 2-group Belyi maps to arithmetic geometry, we now state the main results.

- implementation of an algorithm to enumerate isomorphism classes of 2-group Belyi maps

Main results

Motivated by the applications of 2-group Belyi maps to arithmetic geometry, we now state the main results.

- implementation of an algorithm to enumerate isomorphism classes of 2-group Belyi maps
- implementation of an algorithm to compute equations for 2-group Belyi maps over finite fields

Main results

Motivated by the applications of 2-group Belyi maps to arithmetic geometry, we now state the main results.

- implementation of an algorithm to enumerate isomorphism classes of 2-group Belyi maps
- implementation of an algorithm to compute equations for 2-group Belyi maps over finite fields
- implementation of a *method* to compute equations for 2-group Belyi maps over number fields

Main results

Motivated by the applications of 2-group Belyi maps to arithmetic geometry, we now state the main results.

- implementation of an algorithm to enumerate isomorphism classes of 2-group Belyi maps
- implementation of an algorithm to compute equations for 2-group Belyi maps over finite fields
- implementation of a *method* to compute equations for 2-group Belyi maps over number fields
- computational and theoretical evidence supporting a conjecture that every 2-group Belyi map is defined over an abelian extension of the rationals

Background

Belyi's theorem

A **Belyi map** is a morphism $\phi\colon X \to \mathbb{P}^1$ of smooth projective algebraic curves over $\mathbb{C}$ that is unramified outside of $\{0, 1, \infty\}$.

A **Belyi map** is a morphism $\phi\colon X \to \mathbb{P}^1$ of smooth projective algebraic curves over $\mathbb{C}$ that is unramified outside of $\{0, 1, \infty\}$.

Theorem (Belyi 1979)

An algebraic curve (smooth projective) X over $\mathbb{C}$ can be defined over a number field if and only if X admits a Belyi map.

2-**group Belyi maps**

Let $\phi \colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

2-group Belyi maps

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

The **genus** of ϕ is the genus of the curve X.

Let $\phi \colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

The **genus** of ϕ is the genus of the curve X.

The **degree** of ϕ is the degree of the field extension

$$K(\mathbb{P}^1) \hookrightarrow K(X).$$

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

The **genus** of ϕ is the genus of the curve X.

The **degree** of ϕ is the degree of the field extension

$$K(\mathbb{P}^1) \hookrightarrow K(X).$$

ϕ is **geometrically Galois** if $K^{\mathrm{al}}(X)$ is a Galois extension.

2-group Belyi maps

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

The **genus** of ϕ is the genus of the curve X.

The **degree** of ϕ is the degree of the field extension

$$K(\mathbb{P}^1) \hookrightarrow K(X).$$

ϕ is **geometrically Galois** if $K^{\mathrm{al}}(X)$ is a Galois extension.

The **monodromy group of** ϕ, $\mathrm{Mon}(\phi)$, is the image of the map

$$\pi_1(\mathbb{P}^1 \setminus \{0, 1, \infty\}, \star) \to S_d$$

obtained by path lifting.

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

The **genus** of ϕ is the genus of the curve X.

The **degree** of ϕ is the degree of the field extension

$$K(\mathbb{P}^1) \hookrightarrow K(X).$$

ϕ is **geometrically Galois** if $K^{\mathrm{al}}(X)$ is a Galois extension.

The **monodromy group of** ϕ, $\mathrm{Mon}(\phi)$, is the image of the map

$$\pi_1(\mathbb{P}^1 \setminus \{0, 1, \infty\}, \star) \to S_d$$

obtained by path lifting.

When ϕ is Galois we can identify $\mathrm{Mon}(\phi)$ with $\mathrm{Gal}(K^{\mathrm{al}}(X) \,|\, K^{\mathrm{al}}(\mathbb{P}^1))$.

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map defined over K.

The **genus** of ϕ is the genus of the curve X.

The **degree** of ϕ is the degree of the field extension

$$K(\mathbb{P}^1) \hookrightarrow K(X).$$

ϕ is **geometrically Galois** if $K^{\mathrm{al}}(X)$ is a Galois extension.

The **monodromy group of** ϕ, $\mathrm{Mon}(\phi)$, is the image of the map

$$\pi_1(\mathbb{P}^1 \setminus \{0, 1, \infty\}, \star) \to S_d$$

obtained by path lifting.

When ϕ is Galois we can identify $\mathrm{Mon}(\phi)$ with $\mathrm{Gal}(K^{\mathrm{al}}(X) \,|\, K^{\mathrm{al}}(\mathbb{P}^1))$.

A 2-**group Belyi map** is a Galois Belyi map with monodromy group a 2-group.

Theorem (Beckmann 1989)

Let $\phi\colon X \to \mathbb{P}^1$ be a Galois Belyi map with monodromy group G. Let p be a prime not dividing $\#G$.

Theorem (Beckmann 1989)

Let $\phi\colon X \to \mathbb{P}^1$ be a Galois Belyi map with monodromy group G.
Let p be a prime not dividing $\#G$.
Then there exists a number field M satisfying the following properties.

Theorem (Beckmann 1989)

Let $\phi\colon X \to \mathbb{P}^1$ be a Galois Belyi map with monodromy group G.
Let p be a prime not dividing $\#G$.
Then there exists a number field M satisfying the following properties.

- *p is unramified in M*
- *ϕ is defined over M*
- *X is defined over M*
- *X has good reduction at all primes $\mathfrak{p}$ of M above p*

Permutation Triples

A **transitive permutation triple of degree** d is a triple

$$\sigma = (\sigma_0, \sigma_1, \sigma_\infty) \in S_d^3$$

such that

- $\sigma_\infty \sigma_1 \sigma_0 = 1$
- σ generates a transitive subgroup of S_d

Permutation Triples

A **transitive permutation triple of degree** d is a triple

$$\sigma = (\sigma_0, \sigma_1, \sigma_\infty) \in S_d^3$$

such that

- $\sigma_\infty \sigma_1 \sigma_0 = 1$
- σ generates a transitive subgroup of S_d

The set of degree d Belyi maps up to isomorphism is in bijection with the set of degree d transitive permutation triples up to **simultaneous conjugation** and the group $\langle \sigma \rangle$ is the monodromy group of ϕ.

Passports

A **passport** $\mathcal{P}$ consists of the data (g, G, λ) where $g \geq 0$ is an integer, $G \leq S_d$ is a transitive subgroup, and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d.

Passports

A **passport** $\mathcal{P}$ consists of the data (g, G, λ) where $g \geq 0$ is an integer, $G \leq S_d$ is a transitive subgroup, and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d.

The **passport of a Belyi map** $\phi : X \to \mathbb{P}^1$ is $(g(X), \mathrm{Mon}(\phi), (\lambda_0, \lambda_1, \lambda_\infty))$ with $g(X)$ the genus of X, $\mathrm{Mon}(\phi)$ the monodromy group of ϕ, and the partitions from ramification.

Passports

A **passport** $\mathcal{P}$ consists of the data (g, G, λ) where $g \geq 0$ is an integer, $G \leq S_d$ is a transitive subgroup, and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d.

The **passport of a Belyi map** $\phi : X \to \mathbb{P}^1$ is $(g(X), \mathrm{Mon}(\phi), (\lambda_0, \lambda_1, \lambda_\infty))$ with $g(X)$ the genus of X, $\mathrm{Mon}(\phi)$ the monodromy group of ϕ, and the partitions from ramification.

The **passport of a permutation triple** σ is $(g(\sigma), \langle\sigma\rangle, \lambda(\sigma))$ where

$$g(\sigma) = 1 - d + (e(\sigma_0) - e(\sigma_1) - e(\sigma_\infty))/2$$

with

$$e(\tau) = d - \#\text{cycles of } \tau,$$

and $\lambda(\sigma)$ is specified by cycle structures.

Passports

A **passport** $\mathcal{P}$ consists of the data (g, G, λ) where $g \geq 0$ is an integer, $G \leq S_d$ is a transitive subgroup, and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d.

The **passport of a Belyi map** $\phi : X \to \mathbb{P}^1$ is $(g(X), \mathrm{Mon}(\phi), (\lambda_0, \lambda_1, \lambda_\infty))$ with $g(X)$ the genus of X, $\mathrm{Mon}(\phi)$ the monodromy group of ϕ, and the partitions from ramification.

The **passport of a permutation triple** σ is $(g(\sigma), \langle\sigma\rangle, \lambda(\sigma))$ where

$$g(\sigma) = 1 - d + (e(\sigma_0) - e(\sigma_1) - e(\sigma_\infty))/2$$

with

$$e(\tau) = d - \#\text{cycles of } \tau,$$

and $\lambda(\sigma)$ is specified by cycle structures.

We now discuss the importance of organizing triples by passport.

Fields of moduli, fields of definition, and passports

Let $\phi \colon X \to \mathbb{P}^1$ be a Belyi map.

Fields of moduli, fields of definition, and passports

Let $\phi \colon X \to \mathbb{P}^1$ be a Belyi map.

A number field K is a **field of definition** of X if X can be defined by polynomial equations over K. Similarly for ϕ.

Fields of moduli, fields of definition, and passports

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map.

A number field K is a **field of definition** of X if X can be defined by polynomial equations over K. Similarly for ϕ.

The **field of moduli** $M(X)$ of X is the fixed field of the field automorphisms $\{\tau \in \mathrm{Aut}(\mathbb{C}) : \tau(X) \simeq X\}$. Similarly for ϕ, $M(\phi)$.

Fields of moduli, fields of definition, and passports

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map.

A number field K is a **field of definition** of X if X can be defined by polynomial equations over K. Similarly for ϕ.

The **field of moduli** $M(X)$ of X is the fixed field of the field automorphisms $\{\tau \in \mathrm{Aut}(\mathbb{C}) : \tau(X) \simeq X\}$. Similarly for ϕ, $M(\phi)$.

The **size** of a passport $\mathcal{P}$ is the number of simultaneous conjugacy classes of permutation triples σ with passport $\mathcal{P}$.

Fields of moduli, fields of definition, and passports

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map.

A number field K is a **field of definition** of X if X can be defined by polynomial equations over K. Similarly for ϕ.

The **field of moduli** $M(X)$ of X is the fixed field of the field automorphisms $\{\tau \in \mathrm{Aut}(\mathbb{C}) : \tau(X) \simeq X\}$. Similarly for ϕ, $M(\phi)$.

The **size** of a passport $\mathcal{P}$ is the number of simultaneous conjugacy classes of permutation triples σ with passport $\mathcal{P}$.

Theorem
Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map with passport $\mathcal{P}$. Then the degree of $M(\phi)$ is bounded by the size of $\mathcal{P}$.

Fields of moduli, fields of definition, and passports

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map.

A number field K is a **field of definition** of X if X can be defined by polynomial equations over K. Similarly for ϕ.

The **field of moduli** $M(X)$ of X is the fixed field of the field automorphisms $\{\tau \in \mathrm{Aut}(\mathbb{C}) : \tau(X) \simeq X\}$. Similarly for ϕ, $M(\phi)$.

The **size** of a passport $\mathcal{P}$ is the number of simultaneous conjugacy classes of permutation triples σ with passport $\mathcal{P}$.

Theorem
Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map with passport $\mathcal{P}$. Then the degree of $M(\phi)$ is bounded by the size of $\mathcal{P}$.

A Belyi map need not be defined over its field of moduli!

Fields of moduli, fields of definition, and passports

Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map.

A number field K is a **field of definition** of X if X can be defined by polynomial equations over K. Similarly for ϕ.

The **field of moduli** $M(X)$ of X is the fixed field of the field automorphisms $\{\tau \in \mathrm{Aut}(\mathbb{C}) : \tau(X) \simeq X\}$. Similarly for ϕ, $M(\phi)$.

The **size** of a passport $\mathcal{P}$ is the number of simultaneous conjugacy classes of permutation triples σ with passport $\mathcal{P}$.

Theorem
Let $\phi\colon X \to \mathbb{P}^1$ be a Belyi map with passport $\mathcal{P}$. Then the degree of $M(\phi)$ is bounded by the size of $\mathcal{P}$.

A Belyi map need not be defined over its field of moduli!

The situation improves, however, in the Galois setting.

The Galois setting

Let $\phi\colon X \to \mathbb{P}^1$ be a *Galois* Belyi map of degree d with monodromy group G corresponding to the permutation triple σ.

The Galois setting

Let $\phi\colon X \to \mathbb{P}^1$ be a *Galois* Belyi map of degree d with monodromy group G corresponding to the permutation triple σ.

Then

- ϕ and X are defined over $M(\phi)$,
- $\#G = d$,
- all cycles of σ_s have the same length for $s \in \{0, 1, \infty\}$,
- and if we let a, b, c be the orders of $\sigma_0, \sigma_1, \sigma_\infty$ respectively, we have

$$g(X) = 1 + \frac{\#G}{2}\left(1 - \frac{1}{a} - \frac{1}{b} - \frac{1}{c}\right).$$

Function fields

Let $\phi\colon X \to \mathbb{P}^1$ be a degree d Belyi map defined over K.

Function fields

Let $\phi\colon X \to \mathbb{P}^1$ be a degree d Belyi map defined over K.

This corresponds to a degree d extension $K(\mathbb{P}^1) \hookrightarrow K(X)$.

Function fields

Let $\phi\colon X \to \mathbb{P}^1$ be a degree d Belyi map defined over K.

This corresponds to a degree d extension $K(\mathbb{P}^1) \hookrightarrow K(X)$.

$K(\mathbb{P}^1) \cong K(x)$ the **rational function field** of K in one variable.

Function fields

Let $\phi\colon X \to \mathbb{P}^1$ be a degree d Belyi map defined over K.

This corresponds to a degree d extension $K(\mathbb{P}^1) \hookrightarrow K(X)$.

$K(\mathbb{P}^1) \cong K(x)$ the **rational function field** of K in one variable.

For $K(X)$ separable, we can write

$$K(x)(\alpha) = \frac{K(x)[y]}{(\mathrm{minpoly}_{\alpha,K(x)}(y))}$$

Function fields

Let $\phi\colon X \to \mathbb{P}^1$ be a degree d Belyi map defined over K.

This corresponds to a degree d extension $K(\mathbb{P}^1) \hookrightarrow K(X)$.

$K(\mathbb{P}^1) \cong K(x)$ the **rational function field** of K in one variable.

For $K(X)$ separable, we can write

$$K(x)(\alpha) = \frac{K(x)[y]}{(\mathrm{minpoly}_{\alpha, K(x)}(y))}$$

Ramification in this setting corresponds to the factorization of ideals $(x), (x-1), (1/x)$ in maximal orders of $K(X)$.

Function fields

Let $\phi\colon X \to \mathbb{P}^1$ be a degree d Belyi map defined over K.

This corresponds to a degree d extension $K(\mathbb{P}^1) \hookrightarrow K(X)$.

$K(\mathbb{P}^1) \cong K(x)$ the **rational function field** of K in one variable.

For $K(X)$ separable, we can write

$$K(x)(\alpha) = \frac{K(x)[y]}{(\mathrm{minpoly}_{\alpha,K(x)}(y))}$$

Ramification in this setting corresponds to the factorization of ideals $(x), (x-1), (1/x)$ in maximal orders of $K(X)$.

The monodromy group in this setting corresponds to field automorphisms of the Galois closure of $K(X)$ fixing $K(x)$.

2-group Belyi maps as iterated quadratic extensions

$$
\begin{tikzcd}
X_r \arrow[d, "2"'] \arrow[ddddr, bend left] & \\
\vdots \arrow[d, "2"'] & \\
X_2 \arrow[d, "2"'] \arrow[ddr, bend left] & \\
X_1 \arrow[dr, "2"'] & \\
& X_0 = \mathbb{P}^1
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
K(X_r) \arrow[d, dash, "2"'] \arrow[ddddr, dash, bend left] & \\
\vdots \arrow[d, dash, "2"'] & \\
K(X_2) \arrow[d, dash, "2"'] \arrow[ddr, dash, bend left] & \\
K(X_1) \arrow[dr, dash, "2"'] & \\
& K(X_0) = K(\mathbb{P}^1)
\end{tikzcd}
$$

Computing permutation triples

Setup

We first define some terminology for permutation triples corresponding to 2-group Belyi maps.

Setup

We first define some terminology for permutation triples corresponding to 2-group Belyi maps.

A 2-**group permutation triple** of degree $d \in \mathbb{Z}_{\geq 1}$ is a triple of permutations $\sigma := (\sigma_0, \sigma_1, \sigma_\infty) \in S_d^3$ satisfying

- $\sigma_\infty \sigma_1 \sigma_0 = \mathrm{id}$;
- $G := \langle \sigma_0, \sigma_1 \rangle$ is a transitive subgroup of S_d; and
- G is a 2-group of order d embedded in S_d via its left regular representation.

Setup

We first define some terminology for permutation triples corresponding to 2-group Belyi maps.

A 2-**group permutation triple** of degree $d \in \mathbb{Z}_{\geq 1}$ is a triple of permutations $\sigma := (\sigma_0, \sigma_1, \sigma_\infty) \in S_d^3$ satisfying

- $\sigma_\infty \sigma_1 \sigma_0 = \mathrm{id}$;
- $G := \langle \sigma_0, \sigma_1 \rangle$ is a transitive subgroup of S_d; and
- G is a 2-group of order d embedded in S_d via its left regular representation.

G is called the **monodromy group of** σ.

Setup

We first define some terminology for permutation triples corresponding to 2-group Belyi maps.

A 2-**group permutation triple** of degree $d \in \mathbb{Z}_{\geq 1}$ is a triple of permutations $\sigma := (\sigma_0, \sigma_1, \sigma_\infty) \in S_d^3$ satisfying

- $\sigma_\infty \sigma_1 \sigma_0 = \mathrm{id}$;
- $G := \langle \sigma_0, \sigma_1 \rangle$ is a transitive subgroup of S_d; and
- G is a 2-group of order d embedded in S_d via its left regular representation.

G is called the **monodromy group of** σ.

We say two degree d 2-group permutation triples σ, σ' are **simultaneously conjugate** if there exists $\tau \in S_d$ such that

$$\sigma^\tau := (\tau^{-1}\sigma_0\tau, \tau^{-1}\sigma_1\tau, \tau^{-1}\sigma_\infty\tau,) = \sigma'$$

Lifting permutation triples

Let σ be a 2-group permutation triple.

Lifting permutation triples

Let σ be a 2-group permutation triple.

A **lift** of σ is a 2-group permutation triple $\widetilde{\sigma} \in S_{2d}^3$ such that $\langle \widetilde{\sigma} \rangle$ is isomorphic to some extension $\widetilde{G}$ of $\mathbb{Z}/2\mathbb{Z}$ by G as in the exact sequence below.

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota} \widetilde{G} \xrightarrow{\pi} \langle \sigma \rangle \longrightarrow 1$$

Lifting permutation triples

Let σ be a 2-group permutation triple.

A **lift** of σ is a 2-group permutation triple $\widetilde{\sigma} \in S_{2d}^3$ such that $\langle \widetilde{\sigma} \rangle$ is isomorphic to some extension $\widetilde{G}$ of $\mathbb{Z}/2\mathbb{Z}$ by G as in the exact sequence below.

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota} \widetilde{G} \xrightarrow{\pi} \langle \sigma \rangle \longrightarrow 1$$

For a 2-group permutation triple σ, we denote the set of lifts of σ by $\mathrm{Lifts}(\sigma)$ and $\mathrm{Lifts}(\sigma)/\sim$ denotes the set of lifts up to simultaneous conjugation.

Algorithm to compute $\mathsf{Lifts}(\sigma)/\sim$

Input: σ a 2-group permutation triple of degree d

Output: $\mathsf{Lifts}(\sigma)/\sim$

Algorithm to compute $\text{Lifts}(\sigma)/\sim$

Input: σ a 2-group permutation triple of degree d
Output: $\text{Lifts}(\sigma)/\sim$

1. Let $G = \langle \sigma \rangle$ and compute representatives of $H^2(G, A)$ where $A := \mathbb{Z}/2\mathbb{Z}$ with the trivial G-module structure

Algorithm to compute $\mathrm{Lifts}(\sigma)/\sim$

Input: σ a 2-group permutation triple of degree d
Output: $\mathrm{Lifts}(\sigma)/\sim$

1. Let $G = \langle \sigma \rangle$ and compute representatives of $H^2(G, A)$ where $A := \mathbb{Z}/2\mathbb{Z}$ with the trivial G-module structure
2. For each $f \in H^2(G, A)$ compute the corresponding extension

$$1 \longrightarrow A \xrightarrow{\iota_f} \widetilde{G}_f \xrightarrow{\pi_f} G \longrightarrow 1$$

Algorithm to compute Lifts(σ)/$\sim$

Input: σ a 2-group permutation triple of degree d
Output: Lifts(σ)/$\sim$

1. Let $G = \langle \sigma \rangle$ and compute representatives of $H^2(G, A)$ where $A := \mathbb{Z}/2\mathbb{Z}$ with the trivial G-module structure
2. For each $f \in H^2(G, A)$ compute the corresponding extension
$$1 \longrightarrow A \xrightarrow{\iota_f} \widetilde{G}_f \xrightarrow{\pi_f} G \longrightarrow 1$$
3. For each extension $\widetilde{G}_f$ compute the set Lifts(σ, f) defined by
$$\left\{ \widetilde{\sigma} : \widetilde{\sigma}_s \in \pi_f^{-1}(\sigma_s) \text{ for } s \in \{0, 1, \infty\},\ \widetilde{\sigma}_\infty \widetilde{\sigma}_1 \widetilde{\sigma}_0 = 1,\ \langle \widetilde{\sigma} \rangle = \widetilde{G}_f \right\}$$

Algorithm to compute Lifts$(\sigma)/\sim$

Input: σ a 2-group permutation triple of degree d
Output: Lifts$(\sigma)/\sim$

1. Let $G = \langle \sigma \rangle$ and compute representatives of $H^2(G, A)$ where $A := \mathbb{Z}/2\mathbb{Z}$ with the trivial G-module structure
2. For each $f \in H^2(G, A)$ compute the corresponding extension

$$1 \longrightarrow A \xrightarrow{\iota_f} \widetilde{G}_f \xrightarrow{\pi_f} G \longrightarrow 1$$

3. For each extension $\widetilde{G}_f$ compute the set Lifts(σ, f) defined by

$$\left\{ \widetilde{\sigma} : \widetilde{\sigma}_s \in \pi_f^{-1}(\sigma_s) \text{ for } s \in \{0, 1, \infty\},\ \widetilde{\sigma}_\infty \widetilde{\sigma}_1 \widetilde{\sigma}_0 = 1,\ \langle \widetilde{\sigma} \rangle = \widetilde{G}_f \right\}$$

4.
$$\text{Lifts}(\sigma) := \bigcup_{f \in H^2(G,A)} \text{Lifts}(\sigma, f)$$

Algorithm to compute Lifts$(\sigma)/\!\sim$

Input: σ a 2-group permutation triple of degree d
Output: Lifts$(\sigma)/\!\sim$

1. Let $G = \langle \sigma \rangle$ and compute representatives of $H^2(G, A)$ where $A := \mathbb{Z}/2\mathbb{Z}$ with the trivial G-module structure
2. For each $f \in H^2(G, A)$ compute the corresponding extension
$$1 \longrightarrow A \xrightarrow{\iota_f} \widetilde{G}_f \xrightarrow{\pi_f} G \longrightarrow 1$$
3. For each extension $\widetilde{G}_f$ compute the set Lifts(σ, f) defined by
$$\left\{ \widetilde{\sigma} : \widetilde{\sigma}_s \in \pi_f^{-1}(\sigma_s) \text{ for } s \in \{0, 1, \infty\},\ \widetilde{\sigma}_\infty \widetilde{\sigma}_1 \widetilde{\sigma}_0 = 1,\ \langle \widetilde{\sigma} \rangle = \widetilde{G}_f \right\}$$
4. $$\mathrm{Lifts}(\sigma) := \bigcup_{f \in H^2(G,A)} \mathrm{Lifts}(\sigma, f)$$
5. Quotient Lifts(σ) by simultaneous conjugation

Example computing $\mathsf{Lifts}(\sigma)/\!\sim$: setup

Let $\sigma = \big((1\,2), \mathrm{id}, (1\,2)\big)$. Then $G = \langle \sigma \rangle = \mathbb{Z}/2\mathbb{Z}$.

Example computing $\mathsf{Lifts}(\sigma)/\sim$: setup

Let $\sigma = \big((1\,2), \mathrm{id}, (1\,2)\big)$. Then $G = \langle\sigma\rangle = \mathbb{Z}/2\mathbb{Z}$.

$\widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ and $\widetilde{G}_2 \cong \mathbb{Z}/4\mathbb{Z}$ with

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_1} \widetilde{G}_1 \xrightarrow{\pi_1} G \longrightarrow 1$$

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_2} \widetilde{G}_2 \xrightarrow{\pi_2} G \longrightarrow 1$$

Example computing $\mathrm{Lifts}(\sigma)/\sim$: setup

Let $\sigma = \big((1\,2), \mathrm{id}, (1\,2)\big)$. Then $G = \langle \sigma \rangle = \mathbb{Z}/2\mathbb{Z}$.

$\widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ and $\widetilde{G}_2 \cong \mathbb{Z}/4\mathbb{Z}$ with

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_1} \widetilde{G}_1 \xrightarrow{\pi_1} G \longrightarrow 1$$

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_2} \widetilde{G}_2 \xrightarrow{\pi_2} G \longrightarrow 1$$

Each map π_1, π_2 pulls back to 4 triples that multiply to id:

Example computing $\mathrm{Lifts}(\sigma)/\sim$: setup

Let $\sigma = \big((1\,2), \mathrm{id}, (1\,2)\big)$. Then $G = \langle \sigma \rangle = \mathbb{Z}/2\mathbb{Z}$.

$\widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ and $\widetilde{G}_2 \cong \mathbb{Z}/4\mathbb{Z}$ with

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_1} \widetilde{G}_1 \xrightarrow{\pi_1} G \longrightarrow 1$$

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_2} \widetilde{G}_2 \xrightarrow{\pi_2} G \longrightarrow 1$$

Each map π_1, π_2 pulls back to 4 triples that multiply to id:
$T_1 = \Big\{((1\,2)(3\,4), \mathrm{id}, (1\,2)(3\,4)), ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3)),$
$((1\,4)(2\,3), \mathrm{id}, (1\,4)(2\,3)), ((1\,4)(2\,3), (1\,3)(2\,4), (1\,2)(3\,4))\Big\}$

Example computing $\mathrm{Lifts}(\sigma)/\sim$: setup

Let $\sigma = \big((1\,2), \mathrm{id}, (1\,2)\big)$. Then $G = \langle \sigma \rangle = \mathbb{Z}/2\mathbb{Z}$.

$\widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ and $\widetilde{G}_2 \cong \mathbb{Z}/4\mathbb{Z}$ with

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_1} \widetilde{G}_1 \xrightarrow{\pi_1} G \longrightarrow 1$$

$$1 \longrightarrow \mathbb{Z}/2\mathbb{Z} \xrightarrow{\iota_2} \widetilde{G}_2 \xrightarrow{\pi_2} G \longrightarrow 1$$

Each map π_1, π_2 pulls back to 4 triples that multiply to id:

$T_1 = \Big\{((1\,2)(3\,4), \mathrm{id}, (1\,2)(3\,4)), ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3)),$
$((1\,4)(2\,3), \mathrm{id}, (1\,4)(2\,3)), ((1\,4)(2\,3), (1\,3)(2\,4), (1\,2)(3\,4))\Big\}$

$T_2 = \Big\{((1\,4\,3\,2), \mathrm{id}, (1\,2\,3\,4)), ((1\,2\,3\,4), (1\,3)(2\,4), (1\,2\,3\,4)),$
$((1\,2\,3\,4), \mathrm{id}, (1\,4\,3\,2)), ((1\,4\,3\,2), (1\,3)(2\,4), (1\,4\,3\,2))\Big\}$

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Example computing $\mathrm{Lifts}(\sigma)/\sim$: action on blocks

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Each triple in T_1 must act on the *blocks* $\left\{\boxed{1\,3}, \boxed{2\,4}\right\}$ corresponding to the permutations in σ.

Example computing $\mathrm{Lifts}(\sigma)/\sim$: action on blocks

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Each triple in T_1 must act on the *blocks* $\left\{\boxed{1\,3}, \boxed{2\,4}\right\}$ corresponding to the permutations in σ.

Let $(\widetilde{\sigma}_0, \widetilde{\sigma}_1, \widetilde{\sigma}_\infty) = ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3))$.

Example computing $\mathrm{Lifts}(\sigma)/\!\sim$: action on blocks

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Each triple in T_1 must act on the *blocks* $\left\{\boxed{1\,3}, \boxed{2\,4}\right\}$ corresponding to the permutations in σ.

Let $(\widetilde{\sigma}_0, \widetilde{\sigma}_1, \widetilde{\sigma}_\infty) = ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3))$.

Note that $\widetilde{\sigma}_0\left(\boxed{1\,3}\right) = \boxed{2\,4}$ and $\widetilde{\sigma}_0\left(\boxed{2\,4}\right) = \boxed{1\,3}$.

Example computing $\mathrm{Lifts}(\sigma)/\!\sim$: action on blocks

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Each triple in T_1 must act on the *blocks* $\left\{\boxed{1\,3}, \boxed{2\,4}\right\}$ corresponding to the permutations in σ.

Let $(\widetilde{\sigma}_0, \widetilde{\sigma}_1, \widetilde{\sigma}_\infty) = ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3))$.

Note that $\widetilde{\sigma}_0\left(\boxed{1\,3}\right) = \boxed{2\,4}$ and $\widetilde{\sigma}_0\left(\boxed{2\,4}\right) = \boxed{1\,3}$.

The induced permutation of $\widetilde{\sigma}_0$ on blocks is $\left(\boxed{1\,3}, \boxed{2\,4}\right)$ which is the same as the permutation $\sigma_0 = (1\,2)$.

Example computing $\mathrm{Lifts}(\sigma)/\sim$: action on blocks

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Each triple in T_1 must act on the *blocks* $\left\{\boxed{1\,3}, \boxed{2\,4}\right\}$ corresponding to the permutations in σ.

Let $(\widetilde{\sigma}_0, \widetilde{\sigma}_1, \widetilde{\sigma}_\infty) = ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3))$.

Note that $\widetilde{\sigma}_0\left(\boxed{1\,3}\right) = \boxed{2\,4}$ and $\widetilde{\sigma}_0\left(\boxed{2\,4}\right) = \boxed{1\,3}$.

The induced permutation of $\widetilde{\sigma}_0$ on blocks is $\left(\boxed{1\,3}, \boxed{2\,4}\right)$ which is the same as the permutation $\sigma_0 = (1\,2)$.

Similarly, $\widetilde{\sigma}_1$ acts as id on blocks and $\widetilde{\sigma}_\infty$ acts as $(1\,2)$ on blocks.

Example computing $\mathrm{Lifts}(\sigma)/\sim$: action on blocks

Choose $\alpha = (1\,3)(2\,4)$ to be the generator of $\iota_1(\mathbb{Z}/2\mathbb{Z})$ in $\widetilde{G}_1$.

Each triple in T_1 must act on the *blocks* $\left\{\boxed{1\,3}, \boxed{2\,4}\right\}$ corresponding to the permutations in σ.

Let $(\widetilde{\sigma}_0, \widetilde{\sigma}_1, \widetilde{\sigma}_\infty) = ((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3))$.

Note that $\widetilde{\sigma}_0\left(\boxed{1\,3}\right) = \boxed{2\,4}$ and $\widetilde{\sigma}_0\left(\boxed{2\,4}\right) = \boxed{1\,3}$.

The induced permutation of $\widetilde{\sigma}_0$ on blocks is $\left(\boxed{1\,3}, \boxed{2\,4}\right)$ which is the same as the permutation $\sigma_0 = (1\,2)$.

Similarly, $\widetilde{\sigma}_1$ acts as id on blocks and $\widetilde{\sigma}_\infty$ acts as $(1\,2)$ on blocks.

Choosing

$$\alpha := (1\ d+1)(2\ d+2)\ldots(d-1\ 2d-1)(d\ 2d)$$

allows us to label blocks by reducing modulo d.

Example computing $\mathrm{Lifts}(\sigma)/\!\sim$: conclude

We currently have triples that multiply to id and have the correct action on blocks, but we only want triples that generate the correct group.

Example computing $\mathrm{Lifts}(\sigma)/\sim$: conclude

We currently have triples that multiply to id and have the correct action on blocks, but we only want triples that generate the correct group.

$$\mathrm{Lifts}(\sigma, \widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}) = \Big\{((1\,2)(3\,4),(1\,3)(2\,4),(1\,4)(2\,3)),((1\,4)(2\,3),(1\,3)(2\,4),(1\,2)(3\,4))\Big\}$$

Example computing $\mathrm{Lifts}(\sigma)/\sim$: conclude

We currently have triples that multiply to id and have the correct action on blocks, but we only want triples that generate the correct group.

$\mathrm{Lifts}(\sigma, \widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}) =$
$\Big\{((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3)), ((1\,4)(2\,3), (1\,3)(2\,4), (1\,2)(3\,4))\Big\}$

$\mathrm{Lifts}(\sigma, \widetilde{G}_2 \cong \mathbb{Z}/4\mathbb{Z}) = T_2 =$
$\Big\{((1\,4\,3\,2), \mathrm{id}, (1\,2\,3\,4)), ((1\,2\,3\,4), (1\,3)(2\,4), (1\,2\,3\,4)),$
$((1\,2\,3\,4), \mathrm{id}, (1\,4\,3\,2)), ((1\,4\,3\,2), (1\,3)(2\,4), (1\,4\,3\,2))\Big\}$

Example computing $\mathrm{Lifts}(\sigma)/\sim$: conclude

We currently have triples that multiply to id and have the correct action on blocks, but we only want triples that generate the correct group.

$$\mathrm{Lifts}(\sigma, \widetilde{G}_1 \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}) = \Big\{((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3)), ((1\,4)(2\,3), (1\,3)(2\,4), (1\,2)(3\,4))\Big\}$$

$$\mathrm{Lifts}(\sigma, \widetilde{G}_2 \cong \mathbb{Z}/4\mathbb{Z}) = T_2 = \Big\{((1\,4\,3\,2), \mathrm{id}, (1\,2\,3\,4)), ((1\,2\,3\,4), (1\,3)(2\,4), (1\,2\,3\,4)), ((1\,2\,3\,4), \mathrm{id}, (1\,4\,3\,2)), ((1\,4\,3\,2), (1\,3)(2\,4), (1\,4\,3\,2))\Big\}$$

Lastly, we quotient by simultaneous conjugation to obtain
$$\mathrm{Lifts}(\sigma)/\sim = \Big\{((1\,2)(3\,4), (1\,3)(2\,4), (1\,4)(2\,3)), ((1\,4\,3\,2), \mathrm{id}, (1\,2\,3\,4)), ((1\,2\,3\,4), (1\,3)(2\,4), (1\,2\,3\,4))\Big\}$$

Bipartite graphs of permutation triples

Now that we can lift permutation triples, we now describe some notation for the bipartite graphs that organize these triples.

Bipartite graphs of permutation triples

Now that we can lift permutation triples, we now describe some notation for the bipartite graphs that organize these triples.

For $i \in \mathbb{Z}_{\geq 1}$ we define the bipartite graph denoted $\mathscr{G}_{2^i}$ with the following node sets.

- $\mathscr{G}_{2^i}^{\text{above}}$: the set of isomorphism classes of 2-group Belyi maps of degree 2^i indexed by 2-group permutation triples $\widetilde{\sigma}$ up to simultaneous conjugation in S_{2^i}
- $\mathscr{G}_{2^i}^{\text{below}}$: the set of isomorphism classes of 2-group Belyi maps of degree 2^{i-1} indexed by 2-group permutation triples σ up to simultaneous conjugation in $S_{2^{i-1}}$

Bipartite graphs of permutation triples

Now that we can lift permutation triples, we now describe some notation for the bipartite graphs that organize these triples.

For $i \in \mathbb{Z}_{\geq 1}$ we define the bipartite graph denoted $\mathscr{G}_{2^i}$ with the following node sets.

- $\mathscr{G}_{2^i}^{\text{above}}$: the set of isomorphism classes of 2-group Belyi maps of degree 2^i indexed by 2-group permutation triples $\widetilde{\sigma}$ up to simultaneous conjugation in S_{2^i}
- $\mathscr{G}_{2^i}^{\text{below}}$: the set of isomorphism classes of 2-group Belyi maps of degree 2^{i-1} indexed by 2-group permutation triples σ up to simultaneous conjugation in $S_{2^{i-1}}$

For every pair of nodes $(\widetilde{\sigma}, \sigma) \in \mathscr{G}_{2^i}^{\text{above}} \times \mathscr{G}_{2^i}^{\text{below}}$ there is an edge between σ and $\widetilde{\sigma}$ if and only if $\widetilde{\sigma}$ is simultaneously conjugate to a lift of σ.

Algorithm to compute $\mathscr{G}_{2^i}$

Input: The bipartite graph $\mathscr{G}_{2^{i-1}}$

Output: The bipartite graph $\mathscr{G}_{2^i}$

Algorithm to compute $\mathscr{G}_{2^i}$

Input: The bipartite graph $\mathscr{G}_{2^{i-1}}$
Output: The bipartite graph $\mathscr{G}_{2^i}$

1. $$\mathrm{Lifts}(\mathscr{G}_{2^{i-1}}) := \bigcup_{\sigma \in \mathscr{G}_{2^{i-1}}^{\text{above}}} \mathrm{Lifts}(\sigma)/\sim$$

Algorithm to compute $\mathscr{G}_{2^i}$

Input: The bipartite graph $\mathscr{G}_{2^{i-1}}$
Output: The bipartite graph $\mathscr{G}_{2^i}$

1. $$\mathrm{Lifts}(\mathscr{G}_{2^{i-1}}) := \bigcup_{\sigma \in \mathscr{G}_{2^{i-1}}^{\text{above}}} \mathrm{Lifts}(\sigma)/\sim$$
2. Quotient $\mathrm{Lifts}(\mathscr{G}_{2^{i-1}})$ by simultaneous conjugation in S_{2^i} to obtain $\mathrm{Lifts}(\mathscr{G}_{2^{i-1}})/\sim$

Algorithm to compute $\mathscr{G}_{2^i}$

Input: The bipartite graph $\mathscr{G}_{2^{i-1}}$
Output: The bipartite graph $\mathscr{G}_{2^i}$

1. $$\mathrm{Lifts}(\mathscr{G}_{2^{i-1}}) := \bigcup_{\sigma \in \mathscr{G}_{2^{i-1}}^{\mathrm{above}}} \mathrm{Lifts}(\sigma)/\sim$$
2. Quotient $\mathrm{Lifts}(\mathscr{G}_{2^{i-1}})$ by simultaneous conjugation in S_{2^i} to obtain $\mathrm{Lifts}(\mathscr{G}_{2^{i-1}})/\sim$
3. Define $\mathscr{G}_{2^i}^{\mathrm{below}} := \mathscr{G}_{2^{i-1}}^{\mathrm{above}}$ and define $\mathscr{G}_{2^i}^{\mathrm{above}}$ by representatives of $\mathrm{Lifts}(\mathscr{G}_{2^{i-1}})/\sim$

Input: The bipartite graph $\mathscr{G}_{2^{i-1}}$

Output: The bipartite graph $\mathscr{G}_{2^i}$

1. $$\text{Lifts}(\mathscr{G}_{2^{i-1}}) := \bigcup_{\sigma \in \mathscr{G}_{2^{i-1}}^{\text{above}}} \text{Lifts}(\sigma)/{\sim}$$
2. Quotient $\text{Lifts}(\mathscr{G}_{2^{i-1}})$ by simultaneous conjugation in S_{2^i} to obtain $\text{Lifts}(\mathscr{G}_{2^{i-1}})/{\sim}$
3. Define $\mathscr{G}_{2^i}^{\text{below}} := \mathscr{G}_{2^{i-1}}^{\text{above}}$ and define $\mathscr{G}_{2^i}^{\text{above}}$ by representatives of $\text{Lifts}(\mathscr{G}_{2^{i-1}})/{\sim}$
4. For every pair $(\widetilde{\sigma}, \sigma) \in \mathscr{G}_{2^i}^{\text{above}} \times \mathscr{G}_{2^i}^{\text{below}}$ place an edge between $\widetilde{\sigma}$ and σ if and only if there is a triple in the equivalence class $[\widetilde{\sigma}] \in \text{Lifts}(\mathscr{G}_{2^{i-1}})/{\sim}$ that is a lift of σ

Theorem (M.)

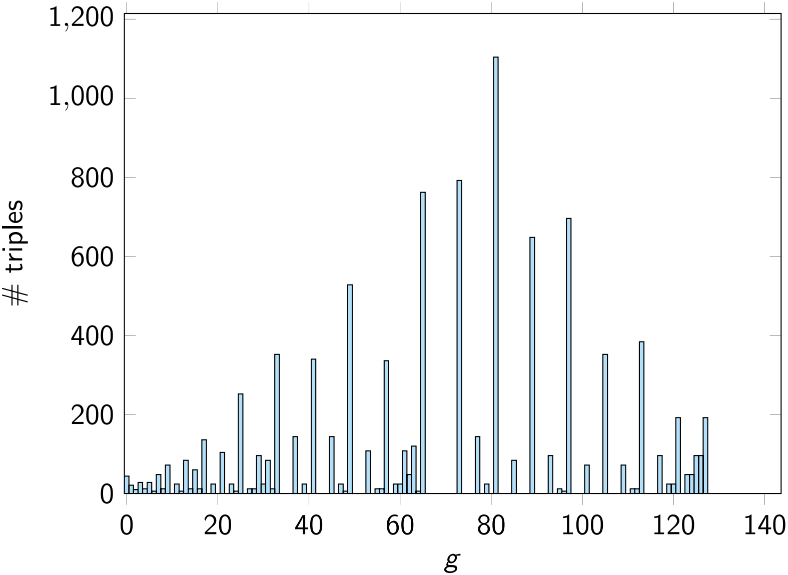
The following tables list the number of isomorphism classes of 2-group Belyi maps, the number of passports, and number of lax passports respectively up to degree 256.

d	1	2	4	8	16	32	64	128	256
# *triples*	1	3	7	19	55	151	503	1799	7175

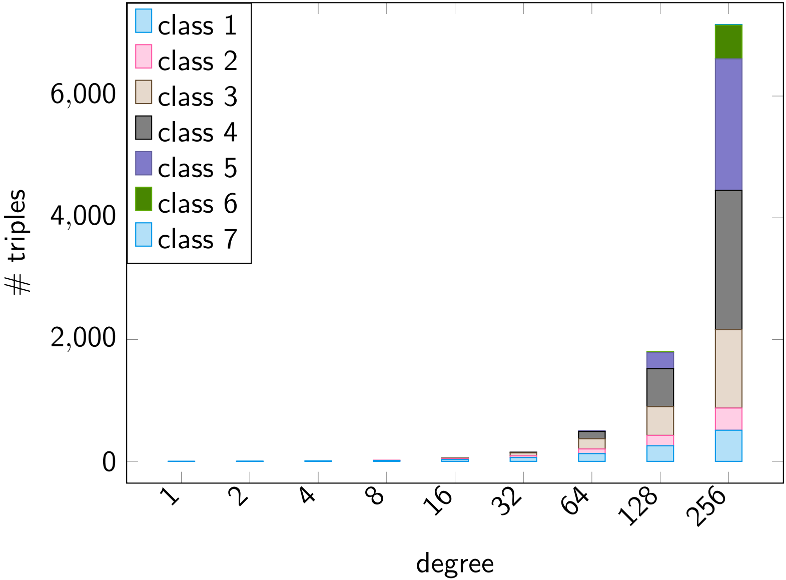
d	1	2	4	8	16	32	64	128	256
# *passports*	1	3	7	16	41	96	267	834	2893

d	1	2	4	8	16	32	64	128	256
# *lax passports*	1	1	3	6	14	31	85	257	882

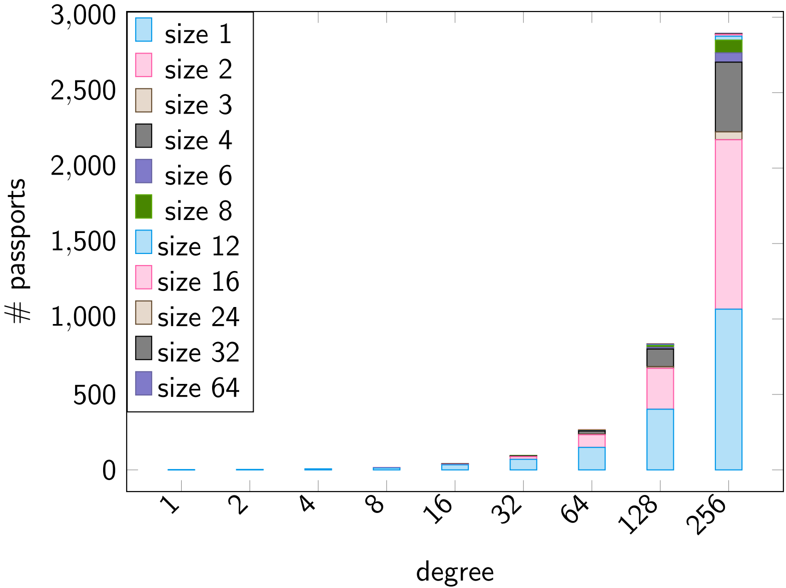
Results : distribution of genera



Results : groups by nilpotency class



Results : passport sizes



A refined conjecture

Passports

Recall that a passport $\mathcal{P}$ consists of the data (g, G, λ) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d corresponding to conjugacy classes (C_0, C_1, C_∞) of S_d.

Passports

Recall that a passport $\mathcal{P}$ consists of the data (g, G, λ) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d corresponding to conjugacy classes (C_0, C_1, C_∞) of S_d.

The size of $\mathcal{P}$ is the cardinality of the set $\Sigma_{\mathcal{P}}$ defined by

$$\Big\{(\sigma_0, \sigma_1, \sigma_\infty) \in C_0 \times C_1 \times C_\infty : \sigma_\infty \sigma_1 \sigma_0 = 1 \text{ and } \langle \sigma_0, \sigma_1 \rangle = G\Big\}/\sim$$

where $\sim$ denotes simultaneous conjugation in S_d.

Passports

Recall that a passport $\mathcal{P}$ consists of the data (g, G, λ) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d corresponding to conjugacy classes (C_0, C_1, C_∞) of S_d.

The size of $\mathcal{P}$ is the cardinality of the set $\Sigma_{\mathcal{P}}$ defined by

$$\Big\{(\sigma_0, \sigma_1, \sigma_\infty) \in C_0 \times C_1 \times C_\infty : \sigma_\infty \sigma_1 \sigma_0 = 1 \text{ and } \langle \sigma_0, \sigma_1 \rangle = G\Big\}/\!\sim$$

where $\sim$ denotes simultaneous conjugation in S_d.

As a result of the action of $G_{\mathbb{Q}}$ on $\mathcal{P}$, the size of $\mathcal{P}$ bounds the degree of the field of moduli of any Belyi map with passport $\mathcal{P}$.

Recall that a passport $\mathcal{P}$ consists of the data (g, G, λ) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $\lambda = (\lambda_0, \lambda_1, \lambda_\infty)$ is a triple of partitions of d corresponding to conjugacy classes (C_0, C_1, C_∞) of S_d.

The size of $\mathcal{P}$ is the cardinality of the set $\Sigma_{\mathcal{P}}$ defined by

$$\Big\{(\sigma_0, \sigma_1, \sigma_\infty) \in C_0 \times C_1 \times C_\infty : \sigma_\infty\sigma_1\sigma_0 = 1 \text{ and } \langle \sigma_0, \sigma_1 \rangle = G\Big\}/\sim$$

where $\sim$ denotes simultaneous conjugation in S_d.

As a result of the action of $G_{\mathbb{Q}}$ on $\mathcal{P}$, the size of $\mathcal{P}$ bounds the degree of the field of moduli of any Belyi map with passport $\mathcal{P}$.

To instead analyze $\mathrm{Gal}(\mathbb{Q}^{\mathrm{al}} \mid \mathbb{Q}^{\mathrm{ab}})$ we *refine* the notion of a passport.

Refined passports

A **refined passport** $\mathscr{P}$ consists of the data (g, G, c) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $c = (c_0, c_1, c_\infty)$ is a triple of conjugacy classes of G.

Refined passports

A **refined passport** $\mathscr{P}$ consists of the data (g, G, c) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $c = (c_0, c_1, c_\infty)$ is a triple of conjugacy classes of G.

The size of $\mathscr{P}$ is the cardinality of the set $\Sigma_{\mathscr{P}}$ defined by

$$\Big\{(\sigma_0, \sigma_1, \sigma_\infty) \in c_0 \times c_1 \times c_\infty : \sigma_\infty \sigma_1 \sigma_0 = 1 \text{ and } \langle \sigma_0, \sigma_1 \rangle = G\Big\}/\sim$$

where $(\sigma_0, \sigma_1, \sigma_\infty) \sim (\sigma_0', \sigma_1', \sigma_\infty')$ if there exists $\alpha \in \mathrm{Aut}(G)$ with $\alpha(\sigma_s) = \sigma_s'$ for every $s \in \{0, 1, \infty\}$.

Refined passports

A **refined passport** $\mathscr{P}$ consists of the data (g, G, c) where $g \in \mathbb{Z}_{\geq 0}$, G is a transitive subgroup of S_d and $c = (c_0, c_1, c_\infty)$ is a triple of conjugacy classes of G.

The size of $\mathscr{P}$ is the cardinality of the set $\Sigma_{\mathscr{P}}$ defined by

$$\Big\{(\sigma_0, \sigma_1, \sigma_\infty) \in c_0 \times c_1 \times c_\infty : \sigma_\infty \sigma_1 \sigma_0 = 1 \text{ and } \langle \sigma_0, \sigma_1 \rangle = G\Big\}/\sim$$

where $(\sigma_0, \sigma_1, \sigma_\infty) \sim (\sigma_0', \sigma_1', \sigma_\infty')$ if there exists $\alpha \in \mathrm{Aut}(G)$ with $\alpha(\sigma_s) = \sigma_s'$ for every $s \in \{0, 1, \infty\}$.

As was the case with passport, every permutation triple σ determines a refined passport $\mathscr{P}(\sigma)$.

A refined conjecture

Theorem (M.)

The size of $\mathscr{P}(\sigma)$ *is equal to* 1 *for every* 2*-group permutation triple* σ *with degree* ≤ 256.

A refined conjecture

Theorem (M.)

The size of $\mathscr{P}(\sigma)$ *is equal to* 1 *for every* 2*-group permutation triple* σ *with degree* ≤ 256.

Conjecture (ARC)

The size of $\mathscr{P}(\sigma)$ *is equal to* 1 *for every* 2*-group permutation triple.*

A refined conjecture

Theorem (M.)

The size of $\mathscr{P}(\sigma)$ *is equal to* 1 *for every* 2*-group permutation triple* σ *with degree* ≤ 256.

Conjecture (ARC)

The size of $\mathscr{P}(\sigma)$ *is equal to* 1 *for every* 2*-group permutation triple.*

Theorem (M.)

ARC is true for 2*-group permutation triples* σ *with* $\langle\sigma\rangle$ *dihedral.*

Computing equations

A motivating example : setup

Let F be a number field with integers $\mathbb{Z}_F$. Let $\mathrm{Pl}(F)$ denote the places of F and S_∞ the archimedean places. For $v \in \mathrm{Pl}(F) \setminus S_\infty$ let $\mathfrak{p}_v$ denote the prime ideal of $\mathbb{Z}_F$ corresponding to v.

Let F be a number field with integers $\mathbb{Z}_F$. Let $\mathrm{Pl}(F)$ denote the places of F and S_∞ the archimedean places. For $v \in \mathrm{Pl}(F) \setminus S_\infty$ let $\mathfrak{p}_v$ denote the prime ideal of $\mathbb{Z}_F$ corresponding to v.

Let $S \subset \mathrm{Pl}(F) \setminus S_\infty$ and let $\mathfrak{a} := \prod_{v \in S} \mathfrak{p}_v$.

Let F be a number field with integers $\mathbb{Z}_F$. Let $\mathrm{Pl}(F)$ denote the places of F and S_∞ the archimedean places. For $v \in \mathrm{Pl}(F) \setminus S_\infty$ let $\mathfrak{p}_v$ denote the prime ideal of $\mathbb{Z}_F$ corresponding to v.

Let $S \subset \mathrm{Pl}(F) \setminus S_\infty$ and let $\mathfrak{a} := \prod_{v \in S} \mathfrak{p}_v$.

Question

How do we construct a quadratic extension of F with ramification prescribed by $\mathfrak{a}$?

A motivating example : setup

Let F be a number field with integers $\mathbb{Z}_F$. Let $\mathrm{Pl}(F)$ denote the places of F and S_∞ the archimedean places. For $v \in \mathrm{Pl}(F) \setminus S_\infty$ let $\mathfrak{p}_v$ denote the prime ideal of $\mathbb{Z}_F$ corresponding to v.

Let $S \subset \mathrm{Pl}(F) \setminus S_\infty$ and let $\mathfrak{a} := \prod_{v \in S} \mathfrak{p}_v$.

Question

How do we construct a quadratic extension of F with ramification prescribed by $\mathfrak{a}$?

First, it is possible that no such extension exists.

A motivating example : setup

Let F be a number field with integers $\mathbb{Z}_F$. Let $\mathrm{Pl}(F)$ denote the places of F and S_∞ the archimedean places. For $v \in \mathrm{Pl}(F) \setminus S_\infty$ let $\mathfrak{p}_v$ denote the prime ideal of $\mathbb{Z}_F$ corresponding to v.

Let $S \subset \mathrm{Pl}(F) \setminus S_\infty$ and let $\mathfrak{a} := \prod_{v \in S} \mathfrak{p}_v$.

Question

How do we construct a quadratic extension of F with ramification prescribed by $\mathfrak{a}$?

First, it is possible that no such extension exists.

Second, if $\mathfrak{a} = (d)$ is principal, then $F(\sqrt{d})$ is ramified exactly at the $\mathfrak{p}_v$, and d is unique up to a unit.

A motivating example : setup

Let F be a number field with integers $\mathbb{Z}_F$. Let $\mathrm{Pl}(F)$ denote the places of F and S_∞ the archimedean places. For $v \in \mathrm{Pl}(F) \setminus S_\infty$ let $\mathfrak{p}_v$ denote the prime ideal of $\mathbb{Z}_F$ corresponding to v.

Let $S \subset \mathrm{Pl}(F) \setminus S_\infty$ and let $\mathfrak{a} := \prod_{v \in S} \mathfrak{p}_v$.

Question

How do we construct a quadratic extension of F with ramification prescribed by $\mathfrak{a}$?

First, it is possible that no such extension exists.

Second, if $\mathfrak{a} = (d)$ is principal, then $F(\sqrt{d})$ is ramified exactly at the $\mathfrak{p}_v$, and d is unique up to a unit.

If $\mathfrak{a}$ is not principal, then the question requires more care.

A motivating example : class group

Let Cl_F denote the class group of F and suppose $\mathfrak{a}\mathfrak{b}^2 = (d)$.

A motivating example : class group

Let Cl_F denote the class group of F and suppose $\mathfrak{a}\mathfrak{b}^2 = (d)$.

Then $[\mathfrak{a}] = [\mathfrak{b}^{-2}]$ implies $\mathfrak{a} \in \mathrm{Cl}_F^2$.

A motivating example : class group

Let Cl_F denote the class group of F and suppose $\mathfrak{a}\mathfrak{b}^2 = (d)$.

Then $[\mathfrak{a}] = [\mathfrak{b}^{-2}]$ implies $\mathfrak{a} \in \mathrm{Cl}_F^2$.

If we let $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$, then $[\mathfrak{a}\mathfrak{b}^2] = [\mathfrak{a}\mathfrak{b}^2][\mathfrak{c}^2] = [\mathfrak{a}(\mathfrak{b}\mathfrak{c})^2]$.

A motivating example : class group

Let Cl_F denote the class group of F and suppose $\mathfrak{a}\mathfrak{b}^2 = (d)$.

Then $[\mathfrak{a}] = [\mathfrak{b}^{-2}]$ implies $\mathfrak{a} \in \mathrm{Cl}_F^2$.

If we let $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$, then $[\mathfrak{a}\mathfrak{b}^2] = [\mathfrak{a}\mathfrak{b}^2][\mathfrak{c}^2] = [\mathfrak{a}(\mathfrak{b}\mathfrak{c})^2]$.

To summarize, in the case where $\mathfrak{a}$ is not principal but there exists $\mathfrak{b}$ with $\mathfrak{a}\mathfrak{b}^2$ principal we have $[\mathfrak{a}] \in \mathrm{Cl}_F^2$ and $[\mathfrak{b}]$ is unique up to multiplication by $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$.

A motivating example : class group

Let Cl_F denote the class group of F and suppose $\mathfrak{a}\mathfrak{b}^2 = (d)$.

Then $[\mathfrak{a}] = [\mathfrak{b}^{-2}]$ implies $\mathfrak{a} \in \mathrm{Cl}_F^2$.

If we let $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$, then $[\mathfrak{a}\mathfrak{b}^2] = [\mathfrak{a}\mathfrak{b}^2][\mathfrak{c}^2] = [\mathfrak{a}(\mathfrak{b}\mathfrak{c})^2]$.

To summarize, in the case where $\mathfrak{a}$ is not principal but there exists $\mathfrak{b}$ with $\mathfrak{a}\mathfrak{b}^2$ principal we have $[\mathfrak{a}] \in \mathrm{Cl}_F^2$ and $[\mathfrak{b}]$ is unique up to multiplication by $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$.

Given $\mathfrak{a}$ encoding ramification data, we want to find $\mathfrak{b}^2$ and d such that $\mathfrak{a}\mathfrak{b}^2 = (d)$.

A motivating example : class group

Let Cl_F denote the class group of F and suppose $\mathfrak{a}\mathfrak{b}^2 = (d)$.

Then $[\mathfrak{a}] = [\mathfrak{b}^{-2}]$ implies $\mathfrak{a} \in \mathrm{Cl}_F^2$.

If we let $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$, then $[\mathfrak{a}\mathfrak{b}^2] = [\mathfrak{a}\mathfrak{b}^2][\mathfrak{c}^2] = [\mathfrak{a}(\mathfrak{b}\mathfrak{c})^2]$.

To summarize, in the case where $\mathfrak{a}$ is not principal but there exists $\mathfrak{b}$ with $\mathfrak{a}\mathfrak{b}^2$ principal we have $[\mathfrak{a}] \in \mathrm{Cl}_F^2$ and $[\mathfrak{b}]$ is unique up to multiplication by $[\mathfrak{c}] \in \mathrm{Cl}_F[2]$.

Given $\mathfrak{a}$ encoding ramification data, we want to find $\mathfrak{b}^2$ and d such that $\mathfrak{a}\mathfrak{b}^2 = (d)$.

The algorithms in this section rely on transporting this technique to the function field setting.

Algebraic function fields : setup

Let K be a perfect field. An **algebraic function field in one variable over** K is a field extension F over K such that there exists $x \in F$ transcendental over K and $[F : K(x)]$ is finite.

Algebraic function fields : setup

Let K be a perfect field. An **algebraic function field in one variable over** K is a field extension F over K such that there exists $x \in F$ transcendental over K and $[F : K(x)]$ is finite.

K is the **constant field** of F and the **exact constant field** of F is the algebraic closure of K in F.

Algebraic function fields : setup

Let K be a perfect field. An **algebraic function field in one variable over** K is a field extension F over K such that there exists $x \in F$ transcendental over K and $[F : K(x)]$ is finite.

K is the **constant field** of F and the **exact constant field** of F is the algebraic closure of K in F.

As an example, let X be an irreducible affine plane curve (possibly singular) defined by the equation $f(x, y) = 0$ with $f \in K[x, y]$. Then the **function field of** X, denoted $K(X)$ is the field of fractions of the coordinate ring $K[x, y]/(f(x, y))$ of X.

Algebraic function fields : setup

Let K be a perfect field. An **algebraic function field in one variable over** K is a field extension F over K such that there exists $x \in F$ transcendental over K and $[F : K(x)]$ is finite.

K is the **constant field** of F and the **exact constant field** of F is the algebraic closure of K in F.

As an example, let X be an irreducible affine plane curve (possibly singular) defined by the equation $f(x, y) = 0$ with $f \in K[x, y]$. Then the **function field of** X, denoted $K(X)$ is the field of fractions of the coordinate ring $K[x, y]/(f(x, y))$ of X.

A **place** of F is the maximal ideal of some DVR $\mathcal{O}_P$ of F. The valuation on F corresponding to P is denoted ord_P.

Algebraic function fields : setup

Let K be a perfect field. An **algebraic function field in one variable over** K is a field extension F over K such that there exists $x \in F$ transcendental over K and $[F : K(x)]$ is finite.

K is the **constant field** of F and the **exact constant field** of F is the algebraic closure of K in F.

As an example, let X be an irreducible affine plane curve (possibly singular) defined by the equation $f(x, y) = 0$ with $f \in K[x, y]$. Then the **function field of** X, denoted $K(X)$ is the field of fractions of the coordinate ring $K[x, y]/(f(x, y))$ of X.

A **place** of F is the maximal ideal of some DVR $\mathcal{O}_P$ of F. The valuation on F corresponding to P is denoted ord_P.

The set of places of F is denoted $\mathrm{Pl}(F)$ and the **degree** of P is the index $[\mathcal{O}_P/P : K]$ of the **residue class field**.

The **divisor class group** $\mathrm{Div}(F)$ of F is the free abelian group generated by the places of F. A **divisor** $D \in \mathrm{Div}(F)$ is represented by a sum of places $\sum_P a_P P$ and the **degree** of D is $\sum_P a_P \deg(P)$. The set of **degree zero divisors** is denoted $\mathrm{Div}^0(F)$.

The **divisor class group** $\mathrm{Div}(F)$ of F is the free abelian group generated by the places of F. A **divisor** $D \in \mathrm{Div}(F)$ is represented by a sum of places $\sum_P a_P P$ and the **degree** of D is $\sum_P a_P \deg(P)$. The set of **degree zero divisors** is denoted $\mathrm{Div}^0(F)$.

The image of $\mathrm{div}\colon F^\times \to \mathrm{Div}(F)$ defined by $\mathrm{div}(f) = \sum_P \mathrm{ord}_P(f) P$ is the subgroup of **principal divisors** of F denoted $\mathrm{Princ}(F)$.

Algebraic function fields : Picard group and $\mathscr{L}(D)$

The **divisor class group** $\mathrm{Div}(F)$ of F is the free abelian group generated by the places of F. A **divisor** $D \in \mathrm{Div}(F)$ is represented by a sum of places $\sum_P a_P P$ and the **degree** of D is $\sum_P a_P \deg(P)$. The set of **degree zero divisors** is denoted $\mathrm{Div}^0(F)$.

The image of $\mathrm{div}\colon F^\times \to \mathrm{Div}(F)$ defined by $\mathrm{div}(f) = \sum_P \mathrm{ord}_P(f)P$ is the subgroup of **principal divisors** of F denoted $\mathrm{Princ}(F)$.

The **Picard group** of F is $\mathrm{Pic}(F) := \mathrm{Div}(F)/\mathrm{Princ}(F)$.
The **Jacobian** of F is $\mathrm{Pic}^0(F) := \mathrm{Div}^0(F)/\mathrm{Princ}(F)$.

Algebraic function fields : Picard group and $\mathscr{L}(D)$

The **divisor class group** $\mathrm{Div}(F)$ of F is the free abelian group generated by the places of F. A **divisor** $D \in \mathrm{Div}(F)$ is represented by a sum of places $\sum_P a_P P$ and the **degree** of D is $\sum_P a_P \deg(P)$. The set of **degree zero divisors** is denoted $\mathrm{Div}^0(F)$.

The image of $\mathrm{div}\colon F^\times \to \mathrm{Div}(F)$ defined by $\mathrm{div}(f) = \sum_P \mathrm{ord}_P(f) P$ is the subgroup of **principal divisors** of F denoted $\mathrm{Princ}(F)$.

The **Picard group** of F is $\mathrm{Pic}(F) := \mathrm{Div}(F)/\mathrm{Princ}(F)$.
The **Jacobian** of F is $\mathrm{Pic}^0(F) := \mathrm{Div}^0(F)/\mathrm{Princ}(F)$.

There is a partial order on $\mathrm{Div}(F)$ defined by $D_1 \geq D_2$ if $\mathrm{ord}_P(D_1) \geq \mathrm{ord}_P(D_2)$ for every $P \in \mathrm{Pl}(F)$.

Algebraic function fields : Picard group and $\mathscr{L}(D)$

The **divisor class group** $\mathrm{Div}(F)$ of F is the free abelian group generated by the places of F. A **divisor** $D \in \mathrm{Div}(F)$ is represented by a sum of places $\sum_P a_P P$ and the **degree** of D is $\sum_P a_P \deg(P)$. The set of **degree zero divisors** is denoted $\mathrm{Div}^0(F)$.

The image of $\mathrm{div}\colon F^\times \to \mathrm{Div}(F)$ defined by $\mathrm{div}(f) = \sum_P \mathrm{ord}_P(f) P$ is the subgroup of **principal divisors** of F denoted $\mathrm{Princ}(F)$.

The **Picard group** of F is $\mathrm{Pic}(F) := \mathrm{Div}(F)/\mathrm{Princ}(F)$.
The **Jacobian** of F is $\mathrm{Pic}^0(F) := \mathrm{Div}^0(F)/\mathrm{Princ}(F)$.

There is a partial order on $\mathrm{Div}(F)$ defined by $D_1 \geq D_2$ if $\mathrm{ord}_P(D_1) \geq \mathrm{ord}_P(D_2)$ for every $P \in \mathrm{Pl}(F)$.

The **Riemann-Roch space** of a divisor $D \in \mathrm{Div}(F)$ is defined by $\mathscr{L}(D) := \{f \in F : \mathrm{div}(f) + D \geq 0\} \cup \{0\}$.

Lemma

Let $aF^{\times 2}$ be a nontrivial coset of $F^{\times}/F^{\times 2}$ and consider the extension $L := F(\sqrt{a})$. Then a prime P of F is ramified in L if and only if $\operatorname{ord}_P(a)$ is odd.

Lemma

Let $aF^{\times 2}$ be a nontrivial coset of $F^{\times}/F^{\times 2}$ and consider the extension $L := F(\sqrt{a})$. Then a prime P of F is ramified in L if and only if $\operatorname{ord}_P(a)$ is odd.

We now revisit the question of finding a quadratic extension of $F(\sqrt{a})$ of F with ramification prescribed by $R \in \operatorname{Div}(F)$.

Lemma

Let $aF^{\times 2}$ be a nontrivial coset of $F^{\times}/F^{\times 2}$ and consider the extension $L := F(\sqrt{a})$. Then a prime P of F is ramified in L if and only if $\mathrm{ord}_P(a)$ is odd.

We now revisit the question of finding a quadratic extension of $F(\sqrt{a})$ of F with ramification prescribed by $R \in \mathrm{Div}(F)$.

The simple cases are when no such a exists or when R is principal. The last case occurs when there exists D such that $R - 2D = \mathrm{div}(a)$ for some $a \in F$.

Lemma

Let $aF^{\times 2}$ be a nontrivial coset of $F^{\times}/F^{\times 2}$ and consider the extension $L := F(\sqrt{a})$. Then a prime P of F is ramified in L if and only if $\mathrm{ord}_P(a)$ is odd.

We now revisit the question of finding a quadratic extension of $F(\sqrt{a})$ of F with ramification prescribed by $R \in \mathrm{Div}(F)$.

The simple cases are when no such a exists or when R is principal. The last case occurs when there exists D such that $R - 2D = \mathrm{div}(a)$ for some $a \in F$.

As in the number field setting, this implies $R \in 2\,\mathrm{Pic}(F)$ and D is unique up to addition by $T \in \mathrm{Pic}^0(F)[2]$.

Algorithm in characteristic $p \geq 3$: setup

Let F be a function field with field of constants $\mathbb{F}_q$ with $q = p^r$ and p an odd prime. Let $\mathbb{F}_q(x)$ denote the rational function field in the variable x.

Algorithm in characteristic $p \geq 3$: setup

Let F be a function field with field of constants $\mathbb{F}_q$ with $q = p^r$ and p an odd prime. Let $\mathbb{F}_q(x)$ denote the rational function field in the variable x.

A 2-**group Belyi map modulo** q is a Galois extension of function fields $\mathbb{F}_q(x) \hookrightarrow F$ with $[F : \mathbb{F}_q(x)]$ a power of 2 unramified outside $\{0, 1, \infty\}$.

Algorithm in characteristic $p \geq 3$: setup

Let F be a function field with field of constants $\mathbb{F}_q$ with $q = p^r$ and p an odd prime. Let $\mathbb{F}_q(x)$ denote the rational function field in the variable x.

A 2-**group Belyi map modulo** q is a Galois extension of function fields $\mathbb{F}_q(x) \hookrightarrow F$ with $[F : \mathbb{F}_q(x)]$ a power of 2 unramified outside $\{0, 1, \infty\}$.

A Belyi map $\phi\colon X \to \mathbb{P}^1$ over a field of characteristic p where $p \nmid \# \operatorname{Mon}(\phi)$ is called **tame**. The theory of tame Belyi maps is the same as in characteristic zero.

Algorithm in characteristic $p \geq 3$: setup

Let F be a function field with field of constants $\mathbb{F}_q$ with $q = p^r$ and p an odd prime. Let $\mathbb{F}_q(x)$ denote the rational function field in the variable x.

A 2-**group Belyi map modulo** q is a Galois extension of function fields $\mathbb{F}_q(x) \hookrightarrow F$ with $[F : \mathbb{F}_q(x)]$ a power of 2 unramified outside $\{0, 1, \infty\}$.

A Belyi map $\phi \colon X \to \mathbb{P}^1$ over a field of characteristic p where $p \nmid \# \operatorname{Mon}(\phi)$ is called **tame**. The theory of tame Belyi maps is the same as in characteristic zero.

Picard groups are implemented in the tame case.

Algorithm in characteristic $p \geq 3$: Galois test

Input:

- F a Galois extension of $\mathbb{F}_q(x)$
- $\mathrm{Gal}(F \,|\, \mathbb{F}_q(x))$ explicitly given as automorphisms of F
- $a \in F$

Output: True if $F(\sqrt{a})$ is Galois over $\mathbb{F}_q(x)$ and False otherwise

Algorithm in characteristic $p \geq 3$: Galois test

Input:

- F a Galois extension of $\mathbb{F}_q(x)$
- $\mathrm{Gal}(F \,|\, \mathbb{F}_q(x))$ explicitly given as automorphisms of F
- $a \in F$

Output: True if $F(\sqrt{a})$ is Galois over $\mathbb{F}_q(x)$ and False otherwise

- For each generator $\sigma \in \mathrm{Gal}(F \,|\, \mathbb{F}_q(x))$ test if $\sigma(a)/a$ is a square in F
- Return True if $\sigma(a)/a$ is a square in F for all generators σ and otherwise return False

Input:

- F a Galois extension of $\mathbb{F}_q(x)$
- $\mathrm{Gal}(F \mid \mathbb{F}_q(x))$ explicitly given as automorphisms of F
- $a \in F$

Output: True if $F(\sqrt{a})$ is Galois over $\mathbb{F}_q(x)$ and False otherwise

- For each generator $\sigma \in \mathrm{Gal}(F \mid \mathbb{F}_q(x))$ test if $\sigma(a)/a$ is a square in F
- Return True if $\sigma(a)/a$ is a square in F for all generators σ and otherwise return False

Similarly, we can apply the same test after extending the constant field from $\mathbb{F}_q$ to $\mathbb{F}_{q^2}$.

Algorithm in characteristic $p \geq 3$: get candidates

Input:

- F a 2-group Belyi map modulo q of degree $d = 2^m$ corresponding to a 2-group permutation triple σ
- A passport $\mathcal{P} = (\widetilde{G}, (a, b, c))$ with $\widetilde{G}$ a 2-group of order $2d$ such that there exists a 2-group permutation triple $\widetilde{\sigma}$ with passport $\mathcal{P}$ that is a lift of σ
- $\mathrm{Gal}(F \,|\, \mathbb{F}_q(x)) \cong \langle \sigma \rangle$ explicitly given as automorphisms of F

Output: A list of candidate functions $\{f_i\}$ with each $f_i \in F$ such that $F(\sqrt{f_i})$ is a 2-group Belyi map modulo q with passport $\mathcal{P}$.

Algorithm in characteristic $p \geq 3$: get candidates (steps 1-4)

1. For $s \in \{0, 1, \infty\}$ compute

$$r_s := \begin{cases} 0 & \text{if } \operatorname{order}(\sigma_s) = \operatorname{order}(\widetilde{\sigma}_s) \\ 1 & \text{if } \operatorname{order}(\sigma_s) < \operatorname{order}(\widetilde{\sigma}_s) \end{cases}$$

Algorithm in characteristic $p \geq 3$: get candidates (steps 1-4)

1. For $s \in \{0, 1, \infty\}$ compute

$$r_s := \begin{cases} 0 & \text{if } \operatorname{order}(\sigma_s) = \operatorname{order}(\widetilde{\sigma}_s) \\ 1 & \text{if } \operatorname{order}(\sigma_s) < \operatorname{order}(\widetilde{\sigma}_s) \end{cases}$$

2. Compute

$$R := \sum_{s \in \{0,1,\infty\}} r_s R_s \in \operatorname{Div}(F)$$

   where R_0, R_1, R_∞ are defined to be the supports of $\operatorname{div}(x)$, $\operatorname{div}(x-1)$, and $\operatorname{div}(1/x)$ respectively.

Algorithm in characteristic $p \geq 3$: get candidates (steps 1-4)

1. For $s \in \{0, 1, \infty\}$ compute

$$r_s := \begin{cases} 0 & \text{if } \mathrm{order}(\sigma_s) = \mathrm{order}(\widetilde{\sigma}_s) \\ 1 & \text{if } \mathrm{order}(\sigma_s) < \mathrm{order}(\widetilde{\sigma}_s) \end{cases}$$

2. Compute

$$R := \sum_{s \in \{0,1,\infty\}} r_s R_s \in \mathrm{Div}(F)$$

   where R_0, R_1, R_∞ are defined to be the supports of $\mathrm{div}(x)$, $\mathrm{div}(x-1)$, and $\mathrm{div}(1/x)$ respectively.
3. Compute the abelian group $\mathrm{Pic}(F) = T \oplus \mathbb{Z}$ (with T a finite abelian group) along with a map $\psi\colon \mathrm{Pic}(F) \to \mathrm{Div}(F)$.

Algorithm in characteristic $p \geq 3$: get candidates (steps 1-4)

1. For $s \in \{0, 1, \infty\}$ compute
$$r_s := \begin{cases} 0 & \text{if order}(\sigma_s) = \text{order}(\widetilde{\sigma}_s) \\ 1 & \text{if order}(\sigma_s) < \text{order}(\widetilde{\sigma}_s) \end{cases}$$
2. Compute
$$R := \sum_{s \in \{0,1,\infty\}} r_s R_s \in \mathrm{Div}(F)$$
where R_0, R_1, R_∞ are defined to be the supports of $\mathrm{div}(x)$, $\mathrm{div}(x-1)$, and $\mathrm{div}(1/x)$ respectively.
3. Compute the abelian group $\mathrm{Pic}(F) = T \oplus \mathbb{Z}$ (with T a finite abelian group) along with a map $\psi \colon \mathrm{Pic}(F) \to \mathrm{Div}(F)$.
4. Compute $[R] := \psi^{-1}(R)$.

Algorithm in characteristic $p \geq 3$: get candidates (step 5)

5. For each $a \in \mathrm{Pic}(F)[2]$ compute the following:

Algorithm in characteristic $p \geq 3$: get candidates (step 5)

5. For each $a \in \mathrm{Pic}(F)[2]$ compute the following:
   (a) Let $D_a := \psi(a + [R]/2) \in \mathrm{Div}(F)$.

Algorithm in characteristic $p \geq 3$: get candidates (step 5)

5. For each $a \in \mathrm{Pic}(F)[2]$ compute the following:
   (a) Let $D_a := \psi(a + [R]/2) \in \mathrm{Div}(F)$.
   (b) Compute $\mathscr{L}(R - 2D_a)$.

Algorithm in characteristic $p \geq 3$: get candidates (step 5)

5. For each $a \in \mathrm{Pic}(F)[2]$ compute the following:
   (a) Let $D_a := \psi(a + [R]/2) \in \mathrm{Div}(F)$.
   (b) Compute $\mathscr{L}(R - 2D_a)$.
   (c) If $\mathscr{L}(R - 2D_a)$ has dimension 1, then compute $f_a \in F$ with $\mathrm{div}(f_a)$ generating $\mathscr{L}(R - 2D_a)$ and go to Step 5d Otherwise go to the next $a \in \mathrm{Pic}(F)[2]$.

Algorithm in characteristic $p \geq 3$: get candidates (step 5)

5. For each $a \in \mathrm{Pic}(F)[2]$ compute the following:
   (a) Let $D_a := \psi(a + [R]/2) \in \mathrm{Div}(F)$.
   (b) Compute $\mathscr{L}(R - 2D_a)$.
   (c) If $\mathscr{L}(R - 2D_a)$ has dimension 1, then compute $f_a \in F$ with $\mathrm{div}(f_a)$ generating $\mathscr{L}(R - 2D_a)$ and go to Step 5d Otherwise go to the next $a \in \mathrm{Pic}(F)[2]$.
   (d) Apply Galois test to F, $\mathrm{Gal}(F \,|\, \mathbb{F}_q(x))$, and f_a from Step 5c to see if $F(\sqrt{f_a})$ generates a Galois extension. If $F(\sqrt{f_a})$ is Galois over $\mathbb{F}_q(x)$ then save f_a and go to the next $a \in \mathrm{Pic}(F)[2]$. If $F(\sqrt{f_a})$ is not Galois over $\mathbb{F}_q(x)$, then go to Step 5e.

Algorithm in characteristic $p \geq 3$: get candidates (step 5)

5. For each $a \in \mathrm{Pic}(F)[2]$ compute the following:
   (a) Let $D_a := \psi(a + [R]/2) \in \mathrm{Div}(F)$.
   (b) Compute $\mathscr{L}(R - 2D_a)$.
   (c) If $\mathscr{L}(R - 2D_a)$ has dimension 1, then compute $f_a \in F$ with $\mathrm{div}(f_a)$ generating $\mathscr{L}(R - 2D_a)$ and go to Step 5d Otherwise go to the next $a \in \mathrm{Pic}(F)[2]$.
   (d) Apply Galois test to F, $\mathrm{Gal}(F \mid \mathbb{F}_q(x))$, and f_a from Step 5c to see if $F(\sqrt{f_a})$ generates a Galois extension. If $F(\sqrt{f_a})$ is Galois over $\mathbb{F}_q(x)$ then save f_a and go to the next $a \in \mathrm{Pic}(F)[2]$. If $F(\sqrt{f_a})$ is not Galois over $\mathbb{F}_q(x)$, then go to Step 5e.
   (e) Let F' be the function field F after extending the field of constants $\mathbb{F}_q$ to $\mathbb{F}_{q^2}$. Apply Galois test to F', $\mathrm{Gal}(F' \mid \mathbb{F}_{q^2}(x))$, and f_a (viewed as an element of F') from Step 5c to see if $F'(\sqrt{f_a})$ generates a Galois extension. If $F(\sqrt{f_a})$ is Galois over $\mathbb{F}_{q^2}(x)$ then save f_a. Go to the next $a \in \mathrm{Pic}(F)[2]$.

Algorithm in characteristic $p \geq 3$: get candidates (steps 6-8)

6. Let S be the set of f_a saved in Step 5d. Let S' be the set of f_a saved in Step 5e.

Algorithm in characteristic $p \geq 3$: get candidates (steps 6-8)

6. Let S be the set of f_a saved in Step 5d. Let S' be the set of f_a saved in Step 5e.
7. - If S is nonempty, then for each $f_a \in S$ compute $F(\sqrt{f_a})$,

$$G_a \cong \mathrm{Gal}(F(\sqrt{f_a}) \,|\, \mathbb{F}_q(x)),$$

and let $S'' = \{f_a \in S : G_a \cong \widetilde{G}\}$.

Algorithm in characteristic $p \geq 3$: get candidates (steps 6-8)

6. Let S be the set of f_a saved in Step 5d. Let S' be the set of f_a saved in Step 5e.
7.
   - If S is nonempty, then for each $f_a \in S$ compute $F(\sqrt{f_a})$,

     $$G_a \cong \mathrm{Gal}(F(\sqrt{f_a}) \,|\, \mathbb{F}_q(x)),$$

     and let $S'' = \{f_a \in S : G_a \cong \widetilde{G}\}$.
   - If S is empty, then for each $f_a \in S'$ compute $F'(\sqrt{f_a})$,

     $$G_a \cong \mathrm{Gal}(F'(\sqrt{f_a}) \,|\, \mathbb{F}_{q^2}(x)),$$

     and let $S'' = \{f_a \in S' : G_a \cong \widetilde{G}\}$.

Algorithm in characteristic $p \geq 3$: get candidates (steps 6-8)

6. Let S be the set of f_a saved in Step 5d. Let S' be the set of f_a saved in Step 5e.
7.
   - If S is nonempty, then for each $f_a \in S$ compute $F(\sqrt{f_a})$,

     $$G_a \cong \mathrm{Gal}(F(\sqrt{f_a}) \,|\, \mathbb{F}_q(x)),$$

     and let $S'' = \{f_a \in S : G_a \cong \widetilde{G}\}$.
   - If S is empty, then for each $f_a \in S'$ compute $F'(\sqrt{f_a})$,

     $$G_a \cong \mathrm{Gal}(F'(\sqrt{f_a}) \,|\, \mathbb{F}_{q^2}(x)),$$

     and let $S'' = \{f_a \in S' : G_a \cong \widetilde{G}\}$.
8. Return the list S''

Algorithm in characteristic $p \geq 3$: compute entire passport

After obtaining candidate functions it is a relatively simple process to compute the quadratic extension of function fields and lift automorphisms.

Algorithm in characteristic $p \geq 3$: compute entire passport

After obtaining candidate functions it is a relatively simple process to compute the quadratic extension of function fields and lift automorphisms.

However, one issue with this technique is that it only guarantees that the resulting Belyi map has the desired *passport* and does not allow us to control which *isomorphism class* we get.

Algorithm in characteristic $p \geq 3$: compute entire passport

After obtaining candidate functions it is a relatively simple process to compute the quadratic extension of function fields and lift automorphisms.

However, one issue with this technique is that it only guarantees that the resulting Belyi map has the desired *passport* and does not allow us to control which *isomorphism class* we get.

To recover from this we use isomorphism testing of function fields to determine if we have redundant Belyi maps with a given passport.

Algorithm in characteristic $p \geq 3$: compute entire passport

After obtaining candidate functions it is a relatively simple process to compute the quadratic extension of function fields and lift automorphisms.

However, one issue with this technique is that it only guarantees that the resulting Belyi map has the desired *passport* and does not allow us to control which *isomorphism class* we get.

To recover from this we use isomorphism testing of function fields to determine if we have redundant Belyi maps with a given passport.

Since we know the sizes of passports from our work with permutation triples, we know that we have representatives from every isomorphism class even if we cannot match the Belyi maps to their corresponding permutation triples.

Implementation in characteristic zero

Our inability to compute $\mathrm{Pic}(F)$ for F over a number field requires us to resort to adhoc techniques in this setting.

Implementation in characteristic zero

Our inability to compute $\mathrm{Pic}(F)$ for F over a number field requires us to resort to adhoc techniques in this setting.

The only difference from the tame case is that without $\mathrm{Pic}(F)$, there is no systematic way to find the candidate functions to extract a square root.

Our inability to compute $\mathrm{Pic}(F)$ for F over a number field requires us to resort to adhoc techniques in this setting.

The only difference from the tame case is that without $\mathrm{Pic}(F)$, there is no systematic way to find the candidate functions to extract a square root.

However, we do have access to the ramification points of the Belyi maps and instead use combinations of these points to try to build a candidate function.

Implementation in characteristic zero

Our inability to compute $\mathrm{Pic}(F)$ for F over a number field requires us to resort to adhoc techniques in this setting.

The only difference from the tame case is that without $\mathrm{Pic}(F)$, there is no systematic way to find the candidate functions to extract a square root.

However, we do have access to the ramification points of the Belyi maps and instead use combinations of these points to try to build a candidate function.

Although this implementation does not allow us to compute all 2-group Belyi maps for a given degree, it does work well in practice.

Results

https://github.com/michaelmusty/2GroupDessins

- *all* 2-group Belyi maps modulo 3 up to degree 32
- hundreds of 2-group Belyi maps up to degree 256

Examples

Notation

`DNG-a,b,c-gE-H`

- `D` : degree in $\{2, 4, 8, 16, 32, 64, 128, 256\}$
- `N` : either `T` or `S` identifying group database
- `G` : a positive integer identifying the group
- `a` : ramification index of 0 in $\{2, 4, 8, 16, 32, 64, 128, 256\}$
- `b` : ramification index of 1 in $\{2, 4, 8, 16, 32, 64, 128, 256\}$
- `c` : ramification index of ∞ in $\{2, 4, 8, 16, 32, 64, 128, 256\}$
- `g` : just the letter g
- `E` : the genus in $\mathbb{Z}_{\geq 0}$
- `H` : the hash of the 2-group permutation triple a positive integer

An interesting example

d3ssins

https://michaelmusty.github.io/d3ssins

Future work

- higher degree over $\mathbb{F}_3$
- an algorithm in characteristic zero
- prove ARC for other families of 2-groups
- p-group Belyi maps for p odd
- compute torsion fields

Acknowledgements

- Dave, Tom, Carl, and John
- Sam, Anna, Jeroen, Edgar, Florian, and Richard
- Mary, Jim, Matt, and Nicole

Galois representations

Let X be an irreducible, smooth projective algebraic curve of genus $g \geq 1$ over a number field K. Let $G_K := \mathsf{Gal}(K^{\mathsf{al}} \mid K)$ be the absolute Galois group of K and let $\ell \in \mathbb{Z}$ be prime.

Let $J := \mathsf{Jac}(X)$ be the **Jacobian variety** of X. J is an abelian variety of dimension g.

G_K acts on the ℓ-torsion points $J[\ell](K^{\mathsf{al}}) \cong (\mathbb{Z}/\ell\mathbb{Z})^{2g}$ of X.

This action determines a **mod-ℓ Galois representation**

$$\rho\colon G_K \to \mathsf{Aut}(J[\ell]) \cong \mathsf{GL}_{2g}(\mathbb{Z}/\ell\mathbb{Z}).$$

The geometry of X and the arithmetic of ρ are inimately related. For example, if X has good reduction at a prime $\mathfrak{p}$ above $p \neq \ell$, then $\mathfrak{p}$ will be unramified in the ℓ-**torsion field** $K(J[\ell])$.

Isomorphism of Belyi maps

Let $\phi\colon X \to \mathbb{P}^1$ and $\phi'\colon X' \to \mathbb{P}^1$ be Belyi maps of degree d. ϕ and ϕ' are **isomorphic** (respectively **lax isomorphic**) if the diagrams

$$\begin{tikzcd} X \arrow[rr, "\sim"] \arrow[dr, "\phi"'] & & X' \arrow[dl, "\phi'"] \\ & \mathbb{P}^1 & \end{tikzcd}, \text{ respectively } \begin{tikzcd} X \arrow[r, "\sim"] \arrow[d, "\phi"'] & X' \arrow[d, "\phi'"] \\ \mathbb{P}^1 \arrow[r, "\sim", "\beta"'] & \mathbb{P}^1 \end{tikzcd}$$

commute where $\beta(\{0, 1, \infty\}) = \{0, 1, \infty\}$.