

Nice Galois groups and nasty ones

Russell Miller

Queens College & CUNY Graduate Center

Midwest Computability Seminar
University of Chicago
25 September 2025

(Partially joint work with Jason Block and with Debanjana Kundu.)

Galois groups

We work with a “ground field” F and an algebraic field extension E of F . The *Galois group* $\text{Gal}(E/F)$ is the group of all automorphisms of E that fix F pointwise. In case $E = \overline{F}$ is the algebraic closure of F , this group is the *absolute Galois group of F* , denoted simply by $\text{Gal}(F)$.

Here E and F will both be computable fields, each with domain ω , and a computable field embedding $F \hookrightarrow E$ will make F a subfield of E . Viewing F as its own image within E , this means that F is c.e. but need not be decidable within E . This image will be decidable if F has a *splitting algorithm*, i.e., if irreducibility in $F[X]$ is decidable.

E will often be an infinite algebraic extension of F , in which case $\text{Gal}(E/F)$ may have size continuum (but no larger). To accommodate this, we present $\text{Gal}(E/F)$ as the set of paths through a tree.

Presenting automorphism groups

Let \mathcal{S} be a computable structure, with domain ω , in a relational signature (R_0, R_1, \dots) . We build the *automorphism tree* $T_{\text{Aut}(\mathcal{S})}$ of \mathcal{S} , which by definition contains all nodes $\tau = (\rho \oplus \lambda) \in \omega^{<\omega}$ with any length $n = |\rho| = |\lambda|$ satisfying:

- $\sigma := \rho \cup (\lambda^{-1})$ is a partial injective function; and

Presenting automorphism groups

Let \mathcal{S} be a computable structure, with domain ω , in a relational signature (R_0, R_1, \dots) . We build the *automorphism tree* $T_{\text{Aut}(\mathcal{S})}$ of \mathcal{S} , which by definition contains all nodes $\tau = (\rho \oplus \lambda) \in \omega^{<\omega}$ with any length $n = |\rho| = |\lambda|$ satisfying:

- $\sigma := \rho \cup (\lambda^{-1})$ is a partial injective function; and
- $(\forall i < n)(\forall (x_1, \dots, x_k) \in (\text{dom}(\sigma))^{\text{arity}(R_i)})$

$$\mathcal{S} \models R_i(x_1, \dots, x_k) \iff \mathcal{S} \models R_i(\sigma(x_1), \dots, \sigma(x_k)).$$

In general $T_{\text{Aut}(\mathcal{S})}$ is an infinite-branching computable subtree of $\omega^{<\omega}$, with terminal nodes, whose paths are in bijection with the elements of $\text{Aut}(\mathcal{S})$. For a path P , the corresponding automorphism is

$$f = \{(n, \rho(n)) : (\exists \lambda) (\rho \oplus \lambda) \in P\} \in \text{Aut}(\mathcal{S}).$$

We can compute composition and inversion on these automorphisms. This is a *computable tree presentation* of $\text{Aut}(\mathcal{S})$.

Tree presentations of structures (of size 2^ω)

Next, an abstract definition of *computable tree presentations* T of structures \mathcal{A}_T :

- The domain of \mathcal{A}_T is the set of all paths through a computable tree $T \subseteq \omega^{<\omega}$.
- The signature of \mathcal{A}_T has no relation symbols except $=$, and all function symbols are computed by Turing functionals on the paths.
- If we have a topology in mind for \mathcal{A}_T , it should have as its basic open sets all $\mathcal{U}_\sigma = \{\text{paths } f : \sigma \sqsubset f\}$ with $\sigma \in T$.

This also suggests the usual presentation of the field \mathbb{R} by fast-converging Cauchy sequences, except that \mathbb{R} requires an equivalence relation on paths. \mathbb{R} has only a *computable tree quotient presentation*, with the quotient topology.

$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :

$x_0 \mapsto$ x_0 (& all conjugates of x_0 over F)

$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :

$x_1 \mapsto$

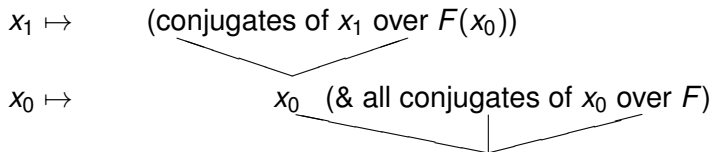
$x_0 \mapsto$

x_0 (& all conjugates of x_0 over F)



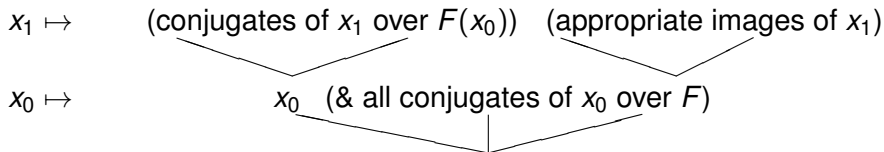
$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



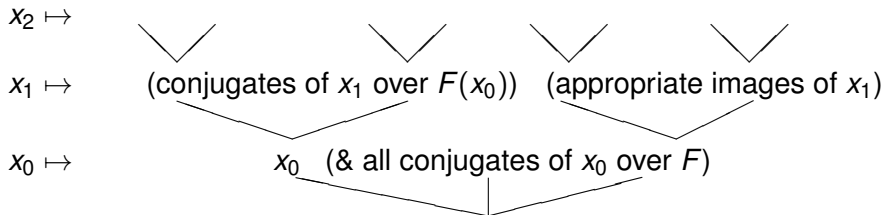
$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



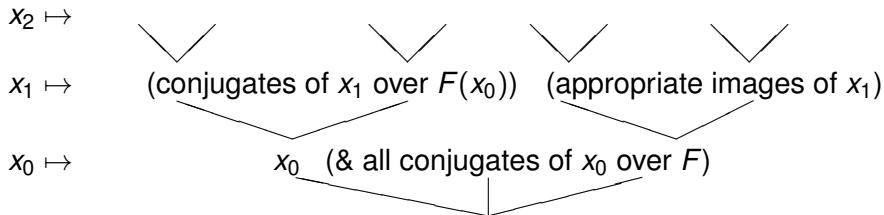
$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

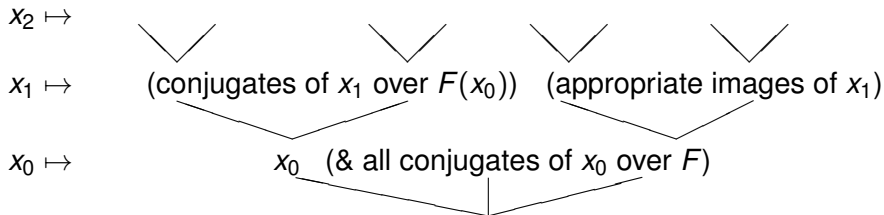
Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



- The paths here will be precisely the automorphisms of E that fix F .

$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

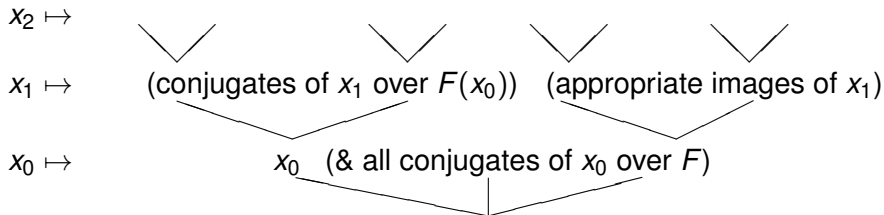
Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



- The paths here will be precisely the automorphisms of E that fix F .
- Since E/F is algebraic, this tree will be finite-branching.

$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

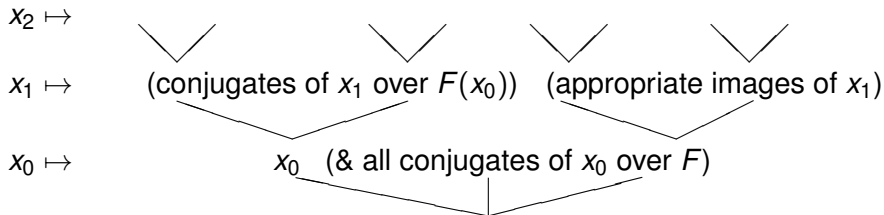
Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



- The paths here will be precisely the automorphisms of E that fix F .
- Since E/F is algebraic, this tree will be finite-branching.
- With a splitting algorithm for F , we can ensure that every $x_i \in F$ maps only to itself, and indeed we can decide F -conjugacy.

$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

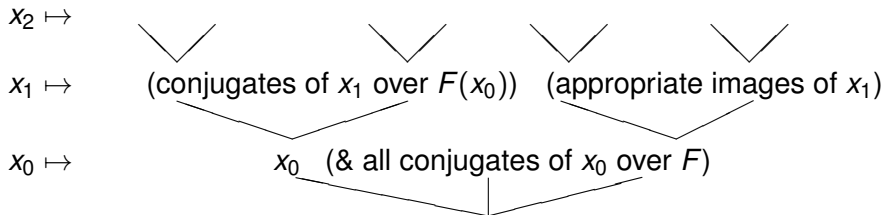
Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



- The paths here will be precisely the automorphisms of E that fix F .
- Since E/F is algebraic, this tree will be finite-branching.
- With a splitting algorithm for F , we can ensure that every $x_i \in F$ maps only to itself, and indeed we can decide F -conjugacy.
- With a splitting algorithm for E as well, (e.g., if $E = \overline{F}$), we can compute the branching function for the tree.

$\text{Gal}(E/F)$ (cf. Calvert-Harizanov-Shlapentokh)

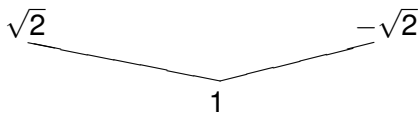
Let E have domain $\{x_0, x_1, x_2, \dots\}$. We ask where to map each x_i :



- The paths here will be precisely the automorphisms of E that fix F .
- Since E/F is algebraic, this tree will be finite-branching.
- With a splitting algorithm for F , we can ensure that every $x_i \in F$ maps only to itself, and indeed we can decide F -conjugacy.
- With a splitting algorithm for E as well, (e.g., if $E = \overline{F}$), we can compute the branching function for the tree.
- Let both have splitting algorithms. Then all nodes extend to paths iff E is “relatively computably categorical over F ” (e.g., if $E = \overline{F}$).

Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$

$\sqrt{2} \mapsto$



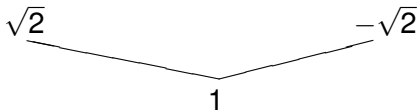
$1 \mapsto$

Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$

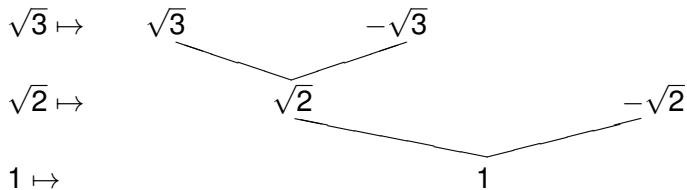
$\sqrt{3} \mapsto$

$\sqrt{2} \mapsto$

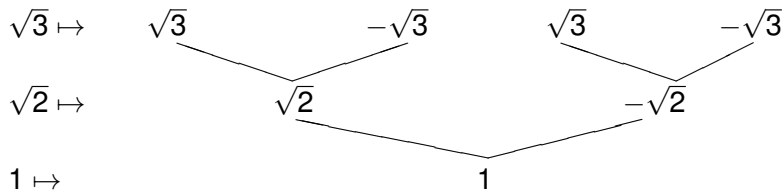
$1 \mapsto$



Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



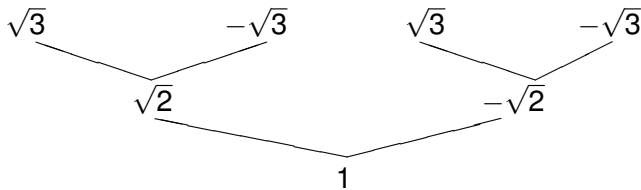
Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$

$\sqrt[4]{6} \mapsto$

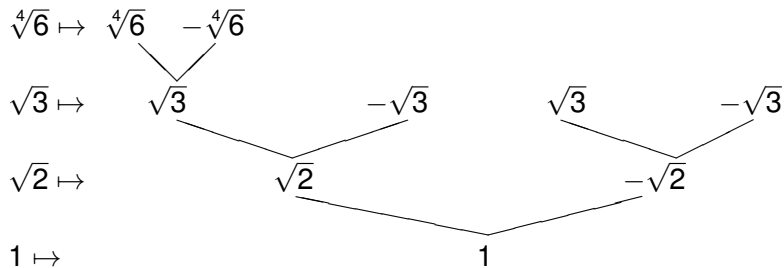
$\sqrt{3} \mapsto$

$\sqrt{2} \mapsto$

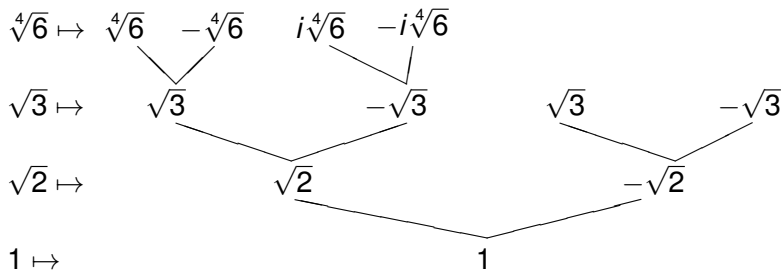
$1 \mapsto$



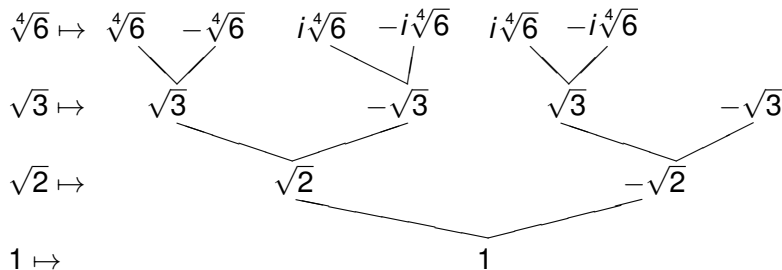
Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



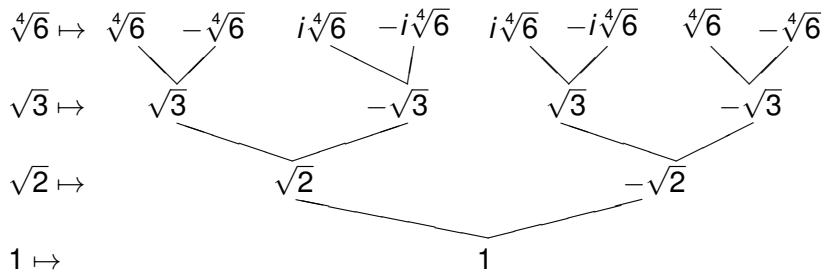
Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



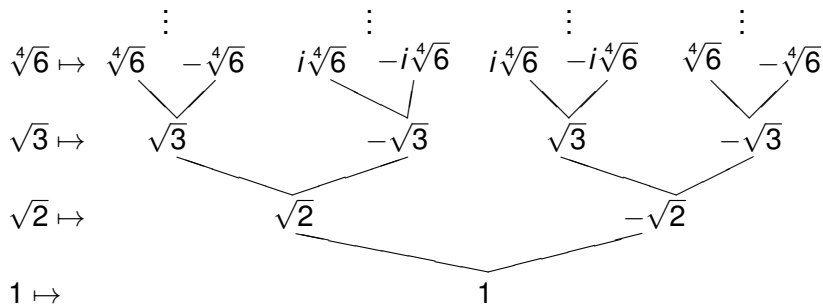
Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



Tree presentation of $\text{Aut}(\overline{\mathbb{Q}}) = \text{Gal}(\mathbb{Q}) = \text{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$



Points to bear in mind:

- At level 1, we simply have the two elements of $\overline{\mathbb{Q}}$ that square to 2. Calling them “positive” and “negative” is arbitrary.
- If we replace $\sqrt[4]{6}$ by a primitive generator of the Galois extension given by $\sqrt{2}$, $\sqrt{3}$, and $\sqrt[4]{6}$, we get a more pleasing picture, with all of its 16 conjugates listed in that row as its images.

Warning: proceed with care

In a (countable) computable structure \mathcal{A} , if one knows that an existential fact $(\exists x) R(x, \vec{a})$ with parameters \vec{a} holds in \mathcal{A} , a simple search procedure, uniform in \vec{a} , is guaranteed to yield a witness $b \in \mathcal{A}$ satisfying $R(b, \vec{a})$.

In $\text{Gal}(E/F)$, even with splitting algorithms for E and F , naming an automorphism f that satisfies a quantifier-free condition $R(f)$ is not so easy, even if you know that one must exist. In fact, $(\exists x) R(x, \vec{a})$ is Σ_1^1 , although it often turns out to be less complex than that.

A *Skolem function*, on input \vec{a} , outputs some witness x to $R(x, \vec{a})$ whenever one exists. Question: can we compute Skolem functions?

The existential theory of $\text{Gal}(E/F)$ is in general arithmetically Σ_2^0 . (Cf. current work by Jason Block.) Counterintuitively, for a *positive* existential sentence with parameters, it is Σ_1^0 for the sentence to be false in $\text{Gal}(E/F)$, and Π_1^0 for it to hold! Notice that mere equality in our presentation is already Π_1^0 -complete.

Computable automorphisms

Definition

For a Turing degree \mathbf{d} , define

$$\text{Gal}_{\mathbf{d}}(E/F) = \{f \in \text{Gal}(E/F) : \deg(f) \leq_T \mathbf{d}\}.$$

So $\text{Gal}_0(E/F)$ is the subgroup of computable automorphisms of E/F .

Computable automorphisms

Definition

For a Turing degree \mathbf{d} , define

$$\mathrm{Gal}_{\mathbf{d}}(E/F) = \{f \in \mathrm{Gal}(E/F) : \deg(f) \leq_T \mathbf{d}\}.$$

So $\mathrm{Gal}_0(E/F)$ is the subgroup of computable automorphisms of E/F .

Question

When is $\mathrm{Gal}_0(E/F)$ an elementary subgroup of $\mathrm{Gal}(E/F)$?

Computable automorphisms

Definition

For a Turing degree \mathbf{d} , define

$$\text{Gal}_{\mathbf{d}}(E/F) = \{f \in \text{Gal}(E/F) : \deg(f) \leq_T \mathbf{d}\}.$$

So $\text{Gal}_0(E/F)$ is the subgroup of computable automorphisms of E/F .

Question

When is $\text{Gal}_0(E/F)$ an elementary subgroup of $\text{Gal}(E/F)$?

For example, let $f \in \text{Gal}_0(E/F)$ be the square of some g in $\text{Gal}(E/F)$. Must there be a computable realization g ? That is, when

$\text{Gal}(E/F) \models (\exists G) G \circ G = f$ and $f \in \text{Gal}_0(E/F)$, does

$\text{Gal}_0(E/F) \models (\exists G) G \circ G = f$ too?

Some specific Galois groups

When F is a finite field and $E = \overline{F}$, we always have

$$\mathrm{Gal}(F) = \mathrm{Gal}(E/F) \cong \widehat{\mathbb{Z}}.$$

Recall $\widehat{\mathbb{Z}} = \prod_{\text{primes } p} \mathbb{Z}_p^+$ is the *profinite completion* of $(\mathbb{Z}, +)$.

Here \mathbb{Z}_p^+ is the additive group of p -adic integers, containing all ω -tuples (a_1, a_2, a_3, \dots) with all $a_n < p^n$ and all $a_{n+1} \equiv a_n \pmod{p^n}$.

For example, with $p = 5$:

$(4, 24, 124, 624, 3124, \dots)$

$(2, 7, 107, 232, 232, 232 + 3125 \cdot 3, \dots)$

$(4, 24, 24, 149, 149, 149, 149, \dots),$

Some specific Galois groups

When F is a finite field and $E = \overline{F}$, we always have

$$\mathrm{Gal}(F) = \mathrm{Gal}(E/F) \cong \widehat{\mathbb{Z}}.$$

Recall $\widehat{\mathbb{Z}} = \prod_{\text{primes } p} \mathbb{Z}_p^+$ is the *profinite completion* of $(\mathbb{Z}, +)$.

Here \mathbb{Z}_p^+ is the additive group of p -adic integers, containing all ω -tuples (a_1, a_2, a_3, \dots) with all $a_n < p^n$ and all $a_{n+1} \equiv a_n \pmod{p^n}$.

For example, with $p = 5$:

$(4, 24, 124, 624, 3124, \dots)$

$(2, 7, 107, 232, 232, 232 + 3125 \cdot 3, \dots)$

$(4, 24, 24, 149, 149, 149, 149, \dots)$, representing the integer 149.

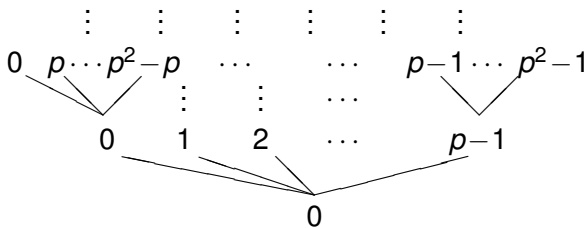
(Indeed, $(4, 24, 124, 624, 3124, \dots)$ above represents the integer -1 .)

\mathbb{Z}_p : the p -adic integers

Again, a p -adic integer is given as a sequence (a_1, a_2, a_3, \dots) with each $a_n \in \mathbb{Z}/p^n\mathbb{Z}$ and $a_{n+1} \equiv a_n \pmod{p^n}$.

These form a ring under coordinatewise $+$ and $\cdot \pmod{p^i}$, with substructure $\mathbb{N} = \{\text{eventually constant sequences}\}$.

We view each sequence (a_0, a_1, \dots) as a path through the complete p -branching tree T_p . $+$ and \cdot are both computable by Turing functionals on those paths:



The additive group \mathbb{Z}_p^+

Here we consider the additive *group* \mathbb{Z}_p^+ , rather than the ring.

First, Block applied work of Szpieg on abelian groups, continued by (Eklof, JSL 1972), to show that for every Turing ideal I , those $g \in \mathbb{Z}_p^+$ with $\deg(g) \in I$ form an elementary subgroup of \mathbb{Z}_p^+ .

Next, consider simple existential formulas in the additive group \mathbb{Z}_p^+ . Equations can always be put in the form $\sum a_i F_i = bG$ with all $a_i, b \in \mathbb{Z}$. When $b \neq 0$, given a tuple \vec{f} from \mathbb{Z}_p^+ , this equation has a solution g in \mathbb{Z}_p^+ iff, for the greatest e such that p^e divides b , $\sum a_i f_i \equiv 0 \pmod{p^e}$. When this holds, g is unique and may be computed uniformly from \vec{f} .

The additive group \mathbb{Z}_p^+

Here we consider the additive *group* \mathbb{Z}_p^+ , rather than the ring.

First, Block applied work of Szpielew on abelian groups, continued by (Eklof, JSL 1972), to show that for every Turing ideal I , those $g \in \mathbb{Z}_p^+$ with $\deg(g) \in I$ form an elementary subgroup of \mathbb{Z}_p^+ .

Next, consider simple existential formulas in the additive group \mathbb{Z}_p^+ . Equations can always be put in the form $\sum a_i F_i = bG$ with all $a_i, b \in \mathbb{Z}$. When $b \neq 0$, given a tuple \vec{f} from \mathbb{Z}_p^+ , this equation has a solution g in \mathbb{Z}_p^+ iff, for the greatest e such that p^e divides b , $\sum a_i f_i \equiv 0 \pmod{p^e}$. When this holds, g is unique and may be computed uniformly from \vec{f} .

However, try computing a Skolem function for the formula

$$(G = 0 \iff F \neq 0).$$

Such a G always exists, but computing one uniformly is impossible!

Decidability and Skolem functions

Disjunctions can fail to have computable Skolem functions, as above. There may be no uniform way to guess which disjunct to try to satisfy. However, every *conjunction* of literals has a computable Skolem function. As an example, given

$$F_1 = G \wedge 3G = 2F_2 - 4F_3,$$

just output f_1 as your g . If this makes the second conjunct false, then no such G exists, so it doesn't matter what you output.

Decidability and Skolem functions

Disjunctions can fail to have computable Skolem functions, as above. There may be no uniform way to guess which disjunct to try to satisfy. However, every *conjunction* of literals has a computable Skolem function. As an example, given

$$F_1 = G \wedge 3G = 2F_2 - 4F_3,$$

just output f_1 as your g . If this makes the second conjunct false, then no such G exists, so it doesn't matter what you output.

In contrast, deciding the set defined by that formula

$$\{(f_1, f_2, f_3) \in (\mathbb{Z}_p^+)^3 : (\exists G) [f_1 = G \wedge 3G = 2f_2 - 4f_3]\}$$

is impossible; we can only enumerate its complement. (Truth of the formula requires $3f_1 = 2f_2 - 4f_3$.)

Reduced existential formulas

There is a procedure for “reducing” a conjunctive existential formula. Above, $(\exists G) [F_1 = G \wedge 3G = 2F_2 - 4F_3]$ would become

$$3F_1 = 2F_2 - 4F_3 \wedge (\exists G) 3G = 2F_2 - 4F_3,$$

where the remaining *reduced* \exists -formula defines a decidable set.

Reduced existential formulas

There is a procedure for “reducing” a conjunctive existential formula. Above, $(\exists G) [F_1 = G \wedge 3G = 2F_2 - 4F_3]$ would become

$$3F_1 = 2F_2 - 4F_3 \wedge (\exists G) 3G = 2F_2 - 4F_3,$$

where the remaining *reduced* \exists -formula defines a decidable set.

Proposition

Every conjunctive sentence with parameters can be expressed as the conjunction of a reduced conjunctive sentence (of the same complexity) and finitely many literals involving only the parameters.

Moreover, every reduced conjunctive sentence defines a decidable set.

The decision procedure extends to all reduced sentences. So Z_p^+ feels like a decidable structure in traditional computable structure theory – except for its atomic diagram: equality remains undecidable!

Tree-decidable structures

Definition (Block-Miller)

Let T be a computable tree presentation of an \mathcal{L} -structure \mathcal{A}_T . We say that T is *tree-decidable* if there exists a Turing functional Φ that decides its elementary diagram in the following sense: whenever $\alpha(X_1, \dots, X_n)$ is a formula from \mathcal{L} , and $(x_1, \dots, x_n) \in [T]^n$ and D is an enumeration of the positive atomic diagram of (x_1, \dots, x_n) in \mathcal{A}_T ,

$$\Phi^{x_1 \oplus \dots \oplus x_n \oplus D}(\ulcorner \alpha(X_1, \dots, X_n) \urcorner) \downarrow = \begin{cases} 1, & \text{if } \mathcal{A}_T \models \alpha(x_1, \dots, x_n); \\ 0, & \text{if } \mathcal{A}_T \models \neg \alpha(x_1, \dots, x_n). \end{cases}$$

So, to decide definable facts about \mathcal{A}_T , we require the formula, the domain elements about which the question is asked, and a list of any nontrivial atomic facts about them. If you want to ask about an (x_1, x_2) with $x_2^{612} = x_1^{493}$, you must eventually disclose this relation.

Tree-decidability of \mathbb{Z}_p^+

Recall: when $b \neq 0$, given a tuple \vec{f} from \mathbb{Z}_p^+ , the equation $\sum a_i F_i = bG$ has a solution g in \mathbb{Z}_p^+ iff, for the greatest e such that p^e divides b , $\sum a_i f_i \equiv 0 \pmod{p^e}$.

So $\{\vec{f} : (\exists G) \sum a_i f_i = bG\}$ is decidable: a finite union of basic open sets at level e in T_p .

Working upwards from this, and reducing formulas by extracting atomic facts about \vec{F} , we get

Proposition (Block-M.)

The tree presentation T_p of \mathbb{Z}_p^+ is tree-decidable.

Tree-decidability of \mathbb{Z}_p^+

Recall: when $b \neq 0$, given a tuple \vec{f} from \mathbb{Z}_p^+ , the equation $\sum a_i F_i = bG$ has a solution g in \mathbb{Z}_p^+ iff, for the greatest e such that p^e divides b , $\sum a_i f_i \equiv 0 \pmod{p^e}$.

So $\{\vec{f} : (\exists G) \sum a_i f_i = bG\}$ is decidable: a finite union of basic open sets at level e in T_p .

Working upwards from this, and reducing formulas by extracting atomic facts about \vec{F} , we get

Proposition (Block-M.)

The tree presentation T_p of \mathbb{Z}_p^+ is tree-decidable.

For $\hat{\mathbb{Z}} = \prod_p \mathbb{Z}_p^+$, given an equation $\sum a_i F_i = bG$ and a \vec{f} , only finitely many p can divide b . We check for these p whether $\sum a_i f_i \equiv 0 \pmod{p^e}$. For all other p , trivially $\sum a_i f_i \equiv 0 \pmod{p^0}$.

So $\{\vec{f} : (\exists G) \sum a_i f_i = bG\}$ is again decidable.

Summarizing the facts about $\widehat{\mathbb{Z}}$

The rest of the proof of tree-decidability of \mathbb{Z}_p^+ carries over to $\widehat{\mathbb{Z}}$. So:

Theorem (Block-M.)

For arbitrary finite fields F , there is a tree presentation of $\text{Gal}(F) \cong \widehat{\mathbb{Z}}$ such that:

- 1 Every conjunctive formula $\alpha(X, Y)$ has a computable Skolem function S_α . This means that, whenever $\text{Gal}(F) \models \exists Y \alpha(x, Y)$, we have $\text{Gal}(F) \models \alpha(x, S_\alpha(x))$.
- 2 $\text{Gal}_0(F)$, containing the computable automorphisms of F , is an elementary subgroup of $\text{Gal}(F)$. (So is every other $\text{Gal}_d(F)$.)
- 3 The presentation is *tree-decidable*, as defined earlier: its elementary diagram is decidable by a Turing functional Φ provided with information about the equality relation on the tuple of parameters given.

Another Galois group

Let C be the cyclotomic field, generated within $\overline{\mathbb{Q}}$ by all primitive m -th roots of unity ζ_m (for all m).

For p prime, $\text{Gal}(\mathbb{Q}(\zeta_{p^n})/\mathbb{Q}) \cong (\mathbb{Z}/p^n\mathbb{Z})^\times$, the multiplicative group of the units in the ring $\mathbb{Z}/p^n\mathbb{Z}$.

When $K_p = \mathbb{Q}(\zeta_p, \zeta_{p^2}, \dots)$, we get $\text{Gal}(K_p/\mathbb{Q}) \cong \mathbb{Z}_p^\times$, the multiplicative group of all units in the ring \mathbb{Z}_p . Here (a_1, a_2, \dots) is a unit iff $a_1 \neq 0$, so the units are the paths through a decidable subtree T_p^\times of the p -branching tree T_p giving \mathbb{Z}_p .

With $p \neq 2$, we get a nicer presentation of \mathbb{Z}_p^\times using the isomorphism $\mathbb{Z}_p^\times \cong (\mathbb{Z}/(p-1)\mathbb{Z})^+ \times \mathbb{Z}_p^+$.

For C itself, $\text{Gal}(C/\mathbb{Q}) \cong \prod_{\text{primes } p} \mathbb{Z}_p^\times$. We present this by “splicing together” the trees T_p^\times into a single tree.

Full picture of $\text{Gal}(\mathbb{C}/\mathbb{Q})$

Putting this together, we can express $\text{Gal}(\mathbb{C}/\mathbb{Q})$ as a product of groups with known tree presentations:

$$\begin{aligned}\text{Gal}(\mathbb{C}/\mathbb{Q}) &\cong \prod_{\text{primes } p} \mathbb{Z}_p^\times \\ &\cong [(\mathbb{Z}/2\mathbb{Z})^+ \times \mathbb{Z}_2^+] \times \prod_{\text{primes } p > 2} ((\mathbb{Z}/(p-1)\mathbb{Z})^+ \times \mathbb{Z}_p^+)\end{aligned}$$

Here every prime p contributes a factor that is finite of even order. In such a group $(\mathbb{Z}/2n\mathbb{Z})^+$, the squares are precisely the even integers.

Full picture of $\text{Gal}(C/\mathbb{Q})$

Putting this together, we can express $\text{Gal}(C/\mathbb{Q})$ as a product of groups with known tree presentations:

$$\begin{aligned}\text{Gal}(C/\mathbb{Q}) &\cong \prod_{\text{primes } p} \mathbb{Z}_p^\times \\ &\cong [(\mathbb{Z}/2\mathbb{Z})^+ \times \mathbb{Z}_2^+] \times \prod_{\text{primes } p > 2} ((\mathbb{Z}/(p-1)\mathbb{Z})^+ \times \mathbb{Z}_p^+)\end{aligned}$$

Here every prime p contributes a factor that is finite of even order. In such a group $(\mathbb{Z}/2n\mathbb{Z})^+$, the squares are precisely the even integers.

Therefore, determining, for an arbitrary $a \in \text{Gal}(C/\mathbb{Q})$, whether $(\exists y) y \cdot y = a$ requires checking that infinitely many coordinates of a be even numbers, because every odd prime p contributes a finite group $(\mathbb{Z}/(p-1)\mathbb{Z})^+$ to $\text{Gal}(C/\mathbb{Q})$. The definable set $\{a \in \text{Gal}(C/\mathbb{Q}) : (\exists y) y \cdot y = a\}$ is Π_1^0 -complete.

Consequences about $\text{Gal}(C/\mathbb{Q})$

Some of the nice properties of $\widehat{\mathbb{Z}}$ still hold in $\prod_p \mathbb{Z}_p^\times$, but not all.
In particular....

Theorem (Block-M.)

There is a tree presentation of $\text{Gal}(C/\mathbb{Q})$ such that:

- 1 Every conjunctive formula $\alpha(X, Y)$ has a computable Skolem function S_α . This means that, whenever $\text{Gal}(F) \models \exists Y \alpha(x, Y)$, we have $\text{Gal}(F) \models \alpha(x, S_\alpha(x))$.
- 2 $\text{Gal}_0(F)$, containing the computable automorphisms of F , is an elementary subgroup of $\text{Gal}(F)$. (So is every other $\text{Gal}_d(F)$.)

However, $\text{Gal}(C/\mathbb{Q})$ has *no* tree-decidable presentation.

Facts about $\text{Gal}(\mathbb{Q})$

We gave a tree presentation of $\text{Gal}(\mathbb{Q})$ earlier.

Theorem (M.)

When I is a *Scott ideal*, the group $\text{Gal}_I(\mathbb{Q})$ of all automorphisms of \mathbb{Q} with Turing degrees in I forms a subgroup of $\text{Gal}(\mathbb{Q})$ that is elementary for all existential and universal formulas and also for all positive formulas.

It is open whether this holds for $\text{Gal}_0(\mathbb{Q})$. We also do not know the arithmetic complexity of definable subsets of $\text{Gal}(F)$. However,....

Theorem (Kundu-M.)

In the standard tree presentation of $\text{Gal}(\mathbb{Q})$, the formula $(\exists Y) Y \cdot Y = X$ has no computable Skolem function.

Skolem functions for $\text{Aut}(\overline{\mathbb{Q}})$

A (generalized) Skolem function for $\text{Aut}(\overline{\mathbb{Q}})$, for the formula $(\exists G) G \circ G = F$, is a function S such that, whenever $f \in \text{Aut}(\overline{\mathbb{Q}})$ satisfies this formula, $S(f) \in \text{Aut}(\overline{\mathbb{Q}})$ with $S(f) \circ S(f) = f$.

Theorem (Kundu-M.)

There is no computable Skolem function for $\text{Aut}(\overline{\mathbb{Q}})$ for the formula $(\exists G) G \circ G = F$.

Proof: Given any Turing functional Φ , run Φ^{id} . If $\Phi^{\text{id} \upharpoonright K_n}(i) \downarrow = \pm i$ for some n , Kundu-M. have a mechanism yielding $f, h \in \text{Aut}(\overline{\mathbb{Q}})$ with

- $f, h \in (\text{Aut}(\overline{\mathbb{Q}}))^2$ with $f \upharpoonright K_n = h \upharpoonright K_n = \text{id} \upharpoonright K_n$.
- Every $g \in \text{Aut}(\overline{\mathbb{Q}})$ with $g \circ g = f$ has $g(i) = i$.
- Every $g \in \text{Aut}(\overline{\mathbb{Q}})$ with $g \circ g = h$ has $g(i) = -i$.

So either $\Phi^f(i)$ or $\Phi^h(i)$ will be incorrect!

The Mechanism

Choose a prime p so large that $\sqrt{p} \notin K_n$. Now $\text{Gal}(\mathbb{Q}(\sqrt[4]{p}, i)/\mathbb{Q}) \cong D_4$, and the permutation $(13)(24)$ of the four conjugates of $\sqrt[4]{p}$ is the square of (1234) and (1432) (and nothing else). Both (1234) and (1432) map i to i . So $(13)(24)$ gives our f , whose square roots all fix i .

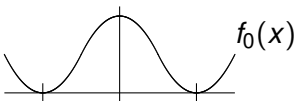
For h , which forces $g(i) = -i$, we use a similar trick involving extensions F containing i that have Galois group S_4 over \mathbb{Q} , hence have $\text{Gal}(F/\mathbb{Q}(i)) \cong A_4$. Here $(13)(24)$ is again the square of (1234) and (1432) and nothing else, and these two 4-cycles are both odd permutations, hence $\notin A_4$, so they both map i to $-i$. S_4 and A_4 are the “generic” Galois groups for degree-4 polynomials over \mathbb{Q} , so it is always possible to find such an extension F with $F \cap K_n = \mathbb{Q}(i)$.

It remains open whether we can repeat this mechanism with other Galois extensions than $\mathbb{Q}(i)$, which would allow us to diagonalize against all computable square roots of an f .

Skolem functions for conjunctive \exists -formulas

In \mathbb{Z}_p^+ , $\widehat{\mathbb{Z}}$, and $\prod_p \mathbb{Z}_p^\times$, all Skolem functions for conjunctive \exists -formulas are computable. In $\text{Gal}(\mathbb{Q})$, they are not. What about other examples?

The square-root function $a \mapsto \sqrt{a}$ is computable in \mathbb{R} on $[0, +\infty)$. However, consider the polynomial $f_a(x) = x^4 - 2x^2 + ax + 1$ with

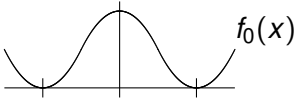
parameter a . Then $f_0(x) = (x^2 - 1)^2$: 

For $a > 0$, $f_a(x) = (x^2 - 1)^2 + ax$ has two roots near -1 , while for $a < 0$, it has two roots near 1 . So no continuous function can produce, for every a , a witness to the statement $(\exists X) f_a(X) = 0$. The enumeration D is necessary for computing this Skolem function.

Skolem functions for conjunctive \exists -formulas

In \mathbb{Z}_p^+ , $\widehat{\mathbb{Z}}$, and $\prod_p \mathbb{Z}_p^\times$, all Skolem functions for conjunctive \exists -formulas are computable. In $\text{Gal}(\mathbb{Q})$, they are not. What about other examples?

The square-root function $a \mapsto \sqrt{a}$ is computable in \mathbb{R} on $[0, +\infty)$. However, consider the polynomial $f_a(x) = x^4 - 2x^2 + ax + 1$ with

parameter a . Then $f_0(x) = (x^2 - 1)^2$: 

For $a > 0$, $f_a(x) = (x^2 - 1)^2 + ax$ has two roots near -1 , while for $a < 0$, it has two roots near 1 . So no continuous function can produce, for every a , a witness to the statement $(\exists X) f_a(X) = 0$. The enumeration D is necessary for computing this Skolem function.

In \mathbb{C} , no Skolem function outputting square roots is computable even when the enumeration D is allowed!

Topics for further study

- Consider the ring \mathbb{Z}_p , rather than just the groups \mathbb{Z}_p^+ and \mathbb{Z}_p^\times . Each individual \mathbb{Z}_p^+ and \mathbb{Z}_p^\times is tree-decidable. Does the ring come (anywhere close to) tree-decidability? What about the field \mathbb{Q}_p ?
- Which standard computable structures have automorphism groups with nice properties? Study $\text{Aut}((\mathbb{Q}, <))$, or $\text{Aut}(\text{the random graph})$. (These are no longer profinite: the trees are ∞ -branching.) Maybe Fraïssé limits in general.
- What properties, if any, must an automorphism group of a computable structure have? Are there computably-tree-presentable groups that are not the automorphism group of any finite structure?
 - ▶ We can ask this for specific classes of computable structures as well: automorphism groups of computable linear orders, or of computable Boolean algebras; Galois groups of field extensions,
- Find a computably-tree-presentable structure in which some \exists -definable set is Σ_1^1 -complete. (Specifically: the set should have a *finite* existential definition.)