# Absolute Galois groups of 2-adic fields, AI, and formalization

David Roe*

Department of Mathematics
Massachusetts Institute of Technology

VaNTAGe seminar

October 27, 2026

*Opus 5.5 created some slides near the end, overall GPT Pro review

# Presentation for $\mathbb{Q}_2$ [RT26, Thm. C.3]

## Theorem

*$\mathrm{Gal}(\overline{\mathbb{Q}}_2/\mathbb{Q}_2)$ is the profinite group generated by $\sigma, \tau, x_0, x_1$ with $\overline{\langle\langle x_0, x_1 \rangle\rangle}$ pro-2 and the following relations*

$$\tau^\sigma = \tau^2$$
$$\langle x_0, \tau \rangle^{-1} x_0^\sigma = x_1^2 [x_1, x_1^{\sigma_2}]$$

$$\sigma_2 = \mathrm{proj}_2(\sigma)$$
$$\langle x_0, \tau \rangle = (x_0^{-3}\tau)^{\mathrm{proj}_2}.$$

Other presentations are also now known or conjectured (see [RT26, Thm. 1.2])

# Presentation for odd $p$ [JW82, Thm. 2]

For $p > 2$, $\mathrm{Gal}(\overline{\mathbb{Q}}_p/\mathbb{Q}_p)$ is the profinite group generated by $\sigma, \tau, x_0, x_1$ with $\overline{\langle\langle x_0, x_1 \rangle\rangle}$ pro-$p$ and the following relations (see [JW82, NSW15])

$$\tau^\sigma = \tau^p$$

$$\langle x_0, \tau \rangle^{-1} x_0^\sigma = x_1^p \Big[ x_1, x_1^{\tau_2^{p+1}} \Big\{ x_1, \tau_2^{p+1} \Big\}^{\sigma_2 \tau_2^{(p-1)/2}} \Big\{ \Big\{ x_1, \tau_2^{p+1} \Big\}, \sigma_2 \tau_2^{(p-1)/2} \Big\}^{\sigma_2 \tau_2^{(p+1)/2} + \tau_2^{(p+1)/2}} \Big]$$

$h \in \mathbb{Z}_p$ with mult. order $p-1$, $\quad \mathrm{proj}_p : \hat{\mathbb{Z}} \to \mathbb{Z}_p$

$$\langle x_0, \tau \rangle := (x_0 \tau x_0^{h^{p-2}} \tau \dots x_0^h \tau)^{\mathrm{proj}_p/(p-1)}$$

$$\beta : \mathrm{Gal}(\mathbb{Q}_p^t/\mathbb{Q}_p) \to \mathbb{Z}_p^\times \qquad \beta(\tau) = h \qquad \beta(\sigma) = 1$$

$$\{x, \rho\} := (x^{\beta(1)} \rho^2 x^{\beta(\rho)} \rho^2 \dots x^{\beta(\rho^{p-2})} \rho^2)^{\mathrm{proj}_p/(p-1)}$$

$$\sigma_2 := \mathrm{proj}_2(\sigma) \qquad \tau_2 := \mathrm{proj}_2(\tau)$$

# Counting algorithm

The number of Galois extensions $L/\mathbb{Q}_p$ with $\mathrm{Gal}(L/\mathbb{Q}_p) \cong G$ is

$$\frac{1}{\#\,\mathrm{Aut}(G)} \#\left\{\varphi : \mathrm{Gal}(\overline{\mathbb{Q}}_p/\mathbb{Q}_p) \twoheadrightarrow G\right\}$$

So it suffices to count the tuples $\sigma, \tau, x_0, x_1 \in G$ (up to automorphism) that

1. satisfy the relations from $\mathrm{Gal}(\overline{\mathbb{Q}}_p/\mathbb{Q}_p)$,
2. have $x_0, x_1 \in O_p(G)$,
3. generate $G$.

## Basic Strategy

Loop over $\sigma$ generating the unramified quotient and $\tau$ generating the tame inertia (with $\tau^\sigma = \tau^p$). For each such $(\sigma, \tau)$ up to automorphism, count the valid $x_0, x_1$ modulo $\mathrm{Stab}_{\mathrm{Aut}(G)}(\sigma, \tau)$.

# Iterative approach

Counting for many $G$, we can build up from quotients.

## Iterative Strategy

- Pick a minimal normal subgroup $N \triangleleft G$, then try to lift $(\sigma, \tau, x_0, x_1)$ from $G/N$ to $G$.
- Tame $G$ form a base case.

Two subtleties.

- If $N$ is not characteristic, it will not be preserved by $\mathrm{Aut}(G)$ so not all automorphisms descend;
- The map $\mathrm{Stab}_{\mathrm{Aut}(G)}(N) \to \mathrm{Aut}(G/N)$ may not be surjective, so equivalent quadruples may become inequivalent.

# Timeline

| | |
|---|---|
| **1982** | Presentation for odd $p$ [JW82] |
| **1984** | Presentation for $K/\mathbb{Q}_2$ with $K(i)/K$ unramified [Die84, p. 152] |
| **2018** | Used odd $p$ presentation in algorithm for counting fields [Roe19] |
| **2023** | $\mathrm{Gal}(\overline{K}/K)$ finitely presented for all $K/\mathbb{Q}_p$ [JS23, Thm. 2.1, 3.4, 4.4] |

2026

| | |
|---|---|
| **Feb** | $\mathrm{Gal}(\overline{\mathbb{Q}}_2/\mathbb{Q}_2)$ in FrontierMath Open Problems |
| **Jun** | Presentations found by Fable 5 and GPT 5.6; informal proof by GPT |
| **Jun–Jul** | Two formalizations down to 9 axioms: one by Fable 5 and Opus 5, one by GPT |
| **Jul–Aug** | Human+AI refinement of writeup using PaperForge |
| **Aug–Oct** | Formalization of 9 axioms in Tau Ceti |
| **Aug–Oct** | Presentation of $\mathrm{Gal}(\overline{K}/K)$ for $K/\mathbb{Q}_2$ |

# Verification

- Wrote a verifier for FrontierMath problem: checks against known tame and pro-2 quotient cases, and against LMFDB data.
- Now transformed into code that uses the presentation to count: github.com/roed314/padicIGP.
- Verifier was an important factor in early success, since agents had a feedback loop.

# Success through AI

- Several failed attempts by GPT 5.4 Pro in March: failed verifier.
- Success shortly after the release of Fable 5 and GPT 5.6.
- Pro harness seems to be qualitatively better.
- Passing the verifier is not a proof: we only check to finite level.
- Output was a 51 page paper: normal AI slop.

# Autoformalization

- 19 day Fable embargo in June led to two independent formalizations.
- Claude formalization finished in 7.25 days (plus axiom-reduction, code-cleanup, alternate presentation).
- Gave confidence in correctness, but stopped short of a full formalization: 9 axioms from class field theory, invariants of quadratic forms, Galois and profinite groups.
- Spent effort aiming to make the resulting Lean code nicer, using Chris Birkbeck's Claude skills.

New infrastructure and metadata: formalization.yaml, Challenge.lean, Palomar, Hexagon.

# Autoformalization infrastructure

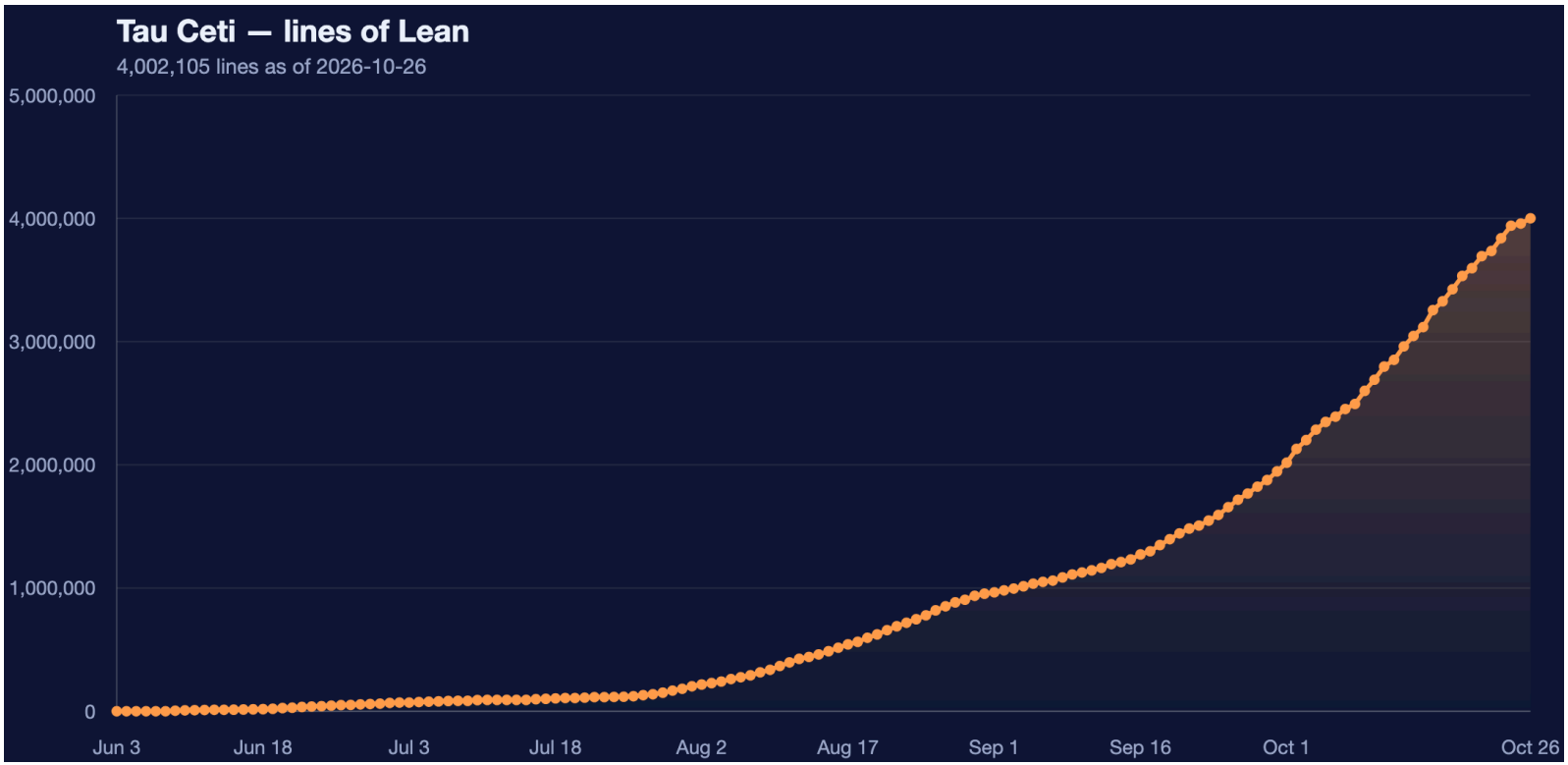
| | |
|---|---|
| `formalization.yaml` | Standard metadata format for formalization projects |
| `Comparator` | Independent verification that a proof is not cheating |
| `Challenge.lean` | Used by `Comparator`, gives formal statement |
| `Palomar` | Public, searchable repository for formal results |
| `Hexagon` | New AI-friendly preprint server |
| `Prove2Me` | Collaborative crowd sourced tool for doing autoformalization |
| `LeanPool` | Automated archive of disconnected formalized results |
| `Tau Ceti` | AI-driven extension of Mathlib, unified library |

# Tau Ceti

- Many autoformalizations start from Mathlib. This leads to duplicated work (e.g. class field theory).
- Mathlib will remain (mostly) human written. Tau Ceti started in July, and is now double the size of Mathlib.
- Tau Ceti is an open, community-driven project building a shared autoformalized library of mathematics in Lean.
- Semantic alignment is a trust problem: centralizing it helps reduce validation effort.
- Working to make focused campaigns easy to run: add the prerequisites for your new result.
- The $\mathrm{Gal}(\overline{\mathbb{Q}}_2/\mathbb{Q}_2)$ campaign finished October 16: formalization is now down to standard 3 axioms (propext, choice, quotient soundness)

# Future of Tau Ceti

taucetiproject.org




- $3 million inference grant from Anthropic
- Efforts to scale much larger: formalize all of mathematics

# PaperForge

Correctness and understanding have been decoupled; we need both. PaperForge aims at the understanding part: take an AI-slop paper and turn it into something humans are excited to read.

## Features

- Incorporates formalization links in the paper
- Reference checking to prevent unintentional plagiarism
- PreText source with latex and html output
- Skills for tasks like explanation of novelty, bridging exposition between results, background writeup

# Quality and open source

## Ground-up coding

AI makes it easy to develop personalized solutions from scratch

## Open source

There are still benefits of open source infrastructure.

- Higher quality code through adversarial review
- Less duplication of effort, wasted resources
- Human community allows for diverse perspectives, synthesizing solutions.

# Proof sketch for $G_{\mathbb{Q}_2}$

Let $\Gamma_A$ be the candidate. Show $\#\mathrm{Sur}(\Gamma_A, G) = \#\mathrm{Sur}(G_{\mathbb{Q}_2}, G)$ for all finite $G$.

1. **Reconstruction.** Build $\Gamma_A$ from its admissible finite quotients, and reduce to equal epimorphism counts.
2. **Marked quotients.** Match the tame quotients, and match the maximal pro-2 quotient with $G_{\mathbb{Q}_2}(2)$, keeping the full unramified character to $\mathbb{Z}_2$.
3. **Fibre product.** Glue the tame and pro-2 data over the common unramified quotient. This leaves a finite lifting problem.
4. **Linear lifting.** For elementary abelian kernels, the Fox calculus on the relators and Galois cohomology give the same obstructions, lift counts, duality and connecting maps.
5. **Quadratic obstruction.** For self-dual simple modules, a word calculation and the square-class filtration give the same Gauss sum.
6. **Induction.** Peel the normal 2-subgroup of $G$ through an elementary quotient and a central Frattini layer. Fourier inversion and strong induction then finish the count.

$$\#\mathrm{Sur}(\Gamma_A, G) = \#\mathrm{Sur}(G_{\mathbb{Q}_2}, G) \ \ \forall G \quad \Longrightarrow \quad \Gamma_A \cong G_{\mathbb{Q}_2}$$

# General $K$

- In addition to $\mathbb{Q}_2$, the presentation of $\mathrm{Gal}(\overline{K}/K)$ also found for $K/\mathbb{Q}_2$ with $K(i)/K$ ramified.
- Formalization to axioms finished September 18, Tau Ceti foundations on October 17.
- AI-slop writeup exists, but need to run it through PaperForge.

# The five families

**Labute type** of the maximal pro-2 quotient of $G_K$, with $n = [K : \mathbb{Q}_2]$:

- $L$ if $n$ is odd.
- $M_\alpha$, relator $a^2[a,b]\,c^{2^\alpha}[c,d]\cdots$, if $n$ is even.
- $N_\alpha$, relator $a^{2+2^\alpha}[a,b]\,[c,d]\cdots$, if $n$ is even.

Here $\alpha \geq 2$ in both even types.
**Unramified level** $r$: $\nu_{\mathrm{ur}}(\ker \chi_{\mathrm{cyc}}) = 2^r\mathbb{Z}_2$.

- Compact: $r = 0$.
- Procyclic: $r \geq 1$, with Frobenius unit $\eta \in \mathbb{Z}_2^\times$.

| Family | Level | Extra data | Quadratic examples |
|---|---|---|---|
| $L$ | $r = 0$ | none | none ($\mathbb{Q}_2$ itself) |
| Compact $N_\alpha$ | $r = 0$ | none | $\mathbb{Q}_2(\sqrt{-2})$ |
| Procyclic $N_\alpha$ | $r \geq 1$ | $\eta$ | none |
| Compact $M_\alpha$ | $r = 0$ | none | $\mathbb{Q}_2(\sqrt{2})$, $\mathbb{Q}_2(\sqrt{5})$ |
| Procyclic $M_\alpha$ | $r \geq 1$ | $\epsilon$, $\eta$ | $\mathbb{Q}_2(\sqrt{10})$, $\mathbb{Q}_2(\sqrt{-10})$ |

Even $\eta$ would force $K(i)/K$ unramified, which the ramified-$i$ hypothesis excludes.

# The admissible group $\Gamma_R$

For a profinite word $R$ in $\sigma, \tau, x_0, \ldots, x_n$:

$$\Gamma_R = \left\langle \sigma, \tau, x_0, \ldots, x_n \;\middle|\; \tau^\sigma = \tau^{q_K},\ R = 1,\ \overline{\langle\langle x_0, \ldots, x_n \rangle\rangle} \text{ pro-}2 \right\rangle_{\text{prof}}$$

- The tame relation is common to all branches.
- Each of the five branches supplies one wild relator $R$.

# Notation

$$x^g = g^{-1}xg, \qquad [x,y] = x^{-1}y^{-1}xy$$

$$\sigma_2 = \sigma^{\omega_2}, \qquad \delta_i = (x_i\tau)^{\omega_2}x_i^{-1}$$

$$m = 2^{\alpha-1}, \qquad s = 2^r, \qquad t = \epsilon\, 2^{r-1}$$

$\hat{\eta} \in \widehat{\mathbb{Z}}^\times$ : 2-component $\eta$, odd components $1$

## Handles

$$H_h^{\mathrm{odd}} = \prod_{j=1}^{h}[x_{2j}, x_{2j+1}] \quad (n = 2h+1), \qquad H_h = \prod_{j=1}^{h}[x_{2j+1}, x_{2j+2}] \quad (n = 2+2h)$$

# Type $L$ and compact $N_\alpha$

## Type $L$, odd degree $n = 2h + 1$

$$R_L = (x_0^{\sigma})^{-1}\,(x_0^{-3}\tau)^{\omega_2}\,x_1^2\,[x_1, x_1^{\sigma_2}]\,H_h^{\mathrm{odd}}$$

## Compact $N_\alpha$, $r = 0$

$$R_{N,\alpha,0} = x_0^{2+2^\alpha}\,[x_0, x_1]\,(x_2^{\sigma})^{-1}\,(x_2\tau)^{\omega_2}\,H_h$$

# Procyclic $N_\alpha$

$r \geq 1$, $\eta = \lambda(-(1+2^\alpha)^{-1}) \in \mathbb{Z}_2^\times$.

$$D_{r,\eta} = \delta_0^{\sigma^{\hat{\eta}}}\, \delta_0^{\sigma^{-2^r}}\, \delta_0^{\sigma^{\hat{\eta}-2^r}}$$

$$R_{N,\alpha,r,\eta} = x_0^{2+2^\alpha}\, [x_0, \sigma^{\hat{\eta}}]\, (x_2^{x_1\sigma^{2^r}})^{-1} (x_2\tau)^{\omega_2} \cdot [D_{r,\eta}, x_1]\, H_h$$

# Compact $M_\alpha$

$r = 0$.

$$A_0 = x_0^{-1}\sigma_2^{-m}$$

$$R_{M,\alpha,0} = A_0^2\,[A_0, x_1]\,\sigma_2^{2^\alpha}\,(x_2^\sigma)^{-1}(x_2\tau)^{\omega_2}$$
$$\cdot\,\delta_1^{\sigma_2^{2m}}\,\delta_1^{\sigma_2^{m}}\,\delta_0^{\sigma_2^{m}}\,\delta_0\,H_h$$

# Procyclic $M_\alpha$: letters

$r \geq 1$, $\eta = \lambda((1-2^\alpha)^{-1})$, $\lambda(-1) = \epsilon\, 2^{r-1}$.

$$C_0 = x_2\sigma_2^s, \quad A = x_0^{-1}C_0^{-m}, \quad B = x_1\sigma_2^t, \quad D = \sigma^{\hat{\eta}}$$

$$\hat{A} = \delta_0^{-1}\sigma_2^{-sm}, \qquad \hat{B} = \delta_1\sigma_2^t$$

$$E_{01} = \delta_1^{\sigma_2^{t+2sm}}\ \delta_1^{\sigma_2^{t+sm}}\ \delta_0^{\sigma_2^{t+sm}}\ \delta_0$$

$$E_2 = \delta_2^{\sigma_2^s} \prod_{j=1}^{m} \left(\delta_2^{\sigma_2^{s(m+j)}}\ \delta_2^{\sigma_2^{t+s(m+j)}}\right)$$

# Procyclic $M_\alpha$: relator

$$R_{\mathrm{lin}} = A^2\,[A,B]\,C_0^{2^\alpha}\,[C_0,D]\,E_{01}\,E_2$$

$$\hat{R} = \hat{A}^2\,[\hat{A},\hat{B}]\,\sigma_2^{s2^\alpha}\,E_{01}$$

$$R_{M,\alpha,r,\epsilon,\eta} = R_{\mathrm{lin}}\,\hat{R}\,\delta_0^2\,[\delta_0,\delta_1]\,H_h$$

$\epsilon = 0$ gives $\mathbb{Q}_2(\sqrt{10})$, and $\epsilon = 1$ gives $\mathbb{Q}_2(\sqrt{-10})$.

# References


[Die84] V. Diekert, Über die absolute Galoisgruppe dyadischer Zahlkörper, *J. Reine Angew. Math.* **350** (1984), 152–172.

[JW82] U. Jannsen and K. Wingberg. Die Struktur der absoluten Galoisgruppe $\mathfrak{p}$-adischer Zahlkörper, *Invent. Math.* **70** (1982), 71–98.

[JS23] M. Jarden and M. Shusterman. The absolute Galois group of a $p$-adic field, *Math. Ann.* **386** (2023), no. 3–4, 1595–1603.

[NSW15] J. Neukirch, A. Schmidt, K. Wingberg. *Cohomology of Number Fields*, 2nd ed., Springer, Berlin, 2015.

[Roe19] D. Roe. The Inverse Galois Problem for p-adic fields, *Proceedings of the Thirteenth Algorithmic Number Theory Symposium (ANTS-XIII)*, Open Book Series 2, Math. Sci. Pub., Berkeley, 2019, pp 393-409

[RT26] D. Roe and D. Turturean. A Presentation of the Absolute Galois Group of $\mathbb{Q}_2$, August 2026, roed314.github.io/gq2.