# Fast Field Trace Computation from Tower of Field Extensions

Julius Zhang

February 3, 2026

## 1 Introduction

This note explains a fast method for computing the field trace map that shows up in lattice-based cryptography. The key idea is that the transitivity of the field trace with respect to field extensions enables logarithmic time (in the degree of the Galois extension) computation of the trace value. This applies to lattice-based cryptographic constructs that operate on the ring of integers $\mathcal{O}_{\mathbb{Q}(\zeta)} \cong \mathbb{Z}[X]/(X^{2^{m-1}}+1)$ (e.g. [NOZ26]), where $\mathbb{Q}(\zeta)$ is some number field formed by adjoining $\zeta$, a primitive $2^m$-th root of unity for some $m \in \mathbb{N}^+$ (we shall assume this notation for $\zeta$ throughout the note). We obtain $\sim 32\times$ speedup in the trace computation for practical parameter choices.

## 2 Background

We recall basic definitions and results in Galois theory about the field trace map. For cryptographic applications, the reader can assume without loss of generality that all field extensions are finite and Galois, which is true for finite extensions over a finite field. For standard references on Galois theory, refer to [DF04].

**Definition 1** (Galois group)**.** *The Galois group $Gal(L/K)$ of a field extension $L/K$ is the group of automorphisms of $L$ that fix $K$.*

**Proposition 1** (Size of the Galois group of a finite field extension)**.** *Let $L/K$ be a finite Galois extension, then $|Gal(L/K)| = [L : K]$.*

There is a field trace map $tr_{L/K} : L \to K$ associated to a finite field extension $L/K$, where computing $tr_{L/K}(x)$ amounts to taking the usual matrix trace over the matrix represented by the endomorphism $\ell_x$ of multiplication by an element $x \in L$. The formula for a finite Galois extension $L/K$ is given by the following result:

**Proposition 2** (Field trace for a Galois extension)**.** *Let $L/K$ be a finite Galois extension of fields. Then the field trace map $tr_{L/K}$ is given by*

$$tr_{L/K}(x) = \sum_{\sigma \in Gal(L/K)} \sigma(x) = [L : K(x)] \sum_{\sigma \in Gal(K(x)/K)} \sigma(x). \tag{1}$$

The field trace has the following computational property:

**Proposition 3** (Transitivity of the field trace)**.** *Let $L/M/K$ be a tower of finite separable field extensions, then the field trace map $tr_{L/K}$ is transitive, i.e.*

$$tr_{L/K}(x) = tr_{M/K} \circ tr_{L/M}(x). \tag{2}$$

*In particular, this holds for a tower of Galois extensions $K \subset M \subset L$.*

# 3 Fast Computation

Proposition 3 allows one to compute the field trace $tr_{L/K}(x)$ efficiently: if one can find a tower of Galois extensions $K = L_1 \subset L_2 \subset \cdots \subset L_n = L$ such that each trace $tr_{L_i/L_{i-1}}$ is easy to compute, then one can recursively compute $tr_{L/K}(x)$ in time logarithmic to the degree $[L : K]$, instead of summing over all the Galois conjugates.

We now show how to apply this to computing the trace map $tr_{\mathbb{Q}(\zeta)/\mathbb{Q}}$. The key technical input uses the structure of the Galois group $\mathrm{Gal}(\mathbb{Q}(\zeta)/\mathbb{Q})$.

**Proposition 4** (Galois group of a cyclotomic extension)**.** *The Galois group $Gal(\mathbb{Q}(\zeta)/\mathbb{Q})$ is isomorphic to the multiplicative subgroup of integers modulo $2^m$, i.e.*

$$Gal(\mathbb{Q}(\zeta)/\mathbb{Q}) \cong (\mathbb{Z}/2^m\mathbb{Z})^\times. \tag{3}$$

**Remark 1.** *A direct generalization of Proposition 4 works for a cyclotomic extension of any degree. For details, see [DF04].*

**Remark 2.** *For $i \in (\mathbb{Z}/2^m\mathbb{Z})^\times$, the corresponding field automorphism acts by $\zeta \mapsto \zeta^i$. In cryptographic literature, this corresponding automorphism is often denoted by $\sigma_i$.*

The following result, due to Gauss, makes the structure of $\mathrm{Gal}(\mathbb{Q}(\zeta)/\mathbb{Q})$ explicit.

**Proposition 5** (Structure of $(\mathbb{Z}/2^m\mathbb{Z})^\times$)**.** *The multiplicative subgroup of integers modulo $2^m$ for $m \geq 2$ is given by*

$$(\mathbb{Z}/2^m\mathbb{Z})^\times \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2^{m-2}\mathbb{Z}, \tag{4}$$

*where the isomorphism in the reverse direction is given by*

$$(a, b) \to (-1)^a 3^b \mod 2^m. \tag{5}$$

We now state the fast trace computation result.

**Proposition 6** (Fast trace computation)**.** *Let $k = 2^t$ be such that $k$ divides $2^{m-2}$, and let $H = \langle \sigma_{-1}, \sigma_{4k+1} \rangle \subset Gal(\mathbb{Q}(\zeta)/\mathbb{Q})$ be the subgroup generated by the automorphisms $\sigma_{-1}$ and $\sigma_{4k+1}$. Let $\mathbb{Q}^H \subset \mathbb{Q}(\zeta)$ be the fixed field of $H$, then we can compute $tr_{\mathbb{Q}(\zeta)/\mathbb{Q}^H}$ by*

$$tr_{\mathbb{Q}(\zeta)/\mathbb{Q}^H}(x) = (1 + \sigma_{-1}) \prod_{j=0}^{m-t-3} (1 + \sigma_{(4k+1)^{2^j}})(x) \tag{6}$$

*Proof.* This essentially follows from the tower of field extensions

$$\{1\} = H_0 \subset H_1 \subset H_2 \subset \cdots \subset H_{m-t-2} \subset H_{m-t-1} = H,$$

where

$$H_j = \langle \sigma_{(4k+1)^{2^{m-t-2-j}}} \rangle \quad \text{for } 1 \leq j \leq m-t-2,$$

and

$$H_{m-t-1} = \langle \sigma_{-1}, \sigma_{4k+1} \rangle = H.$$

To see this, we use the observation that $4k+1 = 2^{t+2}+1$ has order $2^{m-t-2}$ in $(\mathbb{Z}/2^m\mathbb{Z})^\times$.

Each intermediate extension is quadratic with generator given by $\sigma_{(4k+1)^{2^{m-t-2-j}}}$ (or $\sigma_{-1}$ for the final one), so the result follows from Proposition 3.

□

**Corollary 1.** *The tower optimization gives a speedup of $\frac{|H|}{\log_2 |H|}$ compared to the naive approach, where the speedup ratio is counted by the number of field automorphisms.*

# 4 Evaluation

We present some benchmarks of the fast trace computation algorithm. Full code is available at `https://github.com/JuI3s/lattice-crypto`.

| $m$ | $k$ | $\lvert H\rvert$ | Naive ($\mu$s) | Tower ($\mu$s) | Measured | Theoretical |
|---|---|---|---|---|---|---|
| 7 | 4 | 16 | 2.37 | 0.76 | 3.12× | 4.00× |
| 8 | 4 | 32 | 5.84 | 1.19 | 4.91× | 6.40× |
| 9 | 4 | 64 | 19.08 | 1.88 | 10.15× | 10.67× |
| 10 | 4 | 128 | 71.91 | 4.06 | 17.73× | 18.29× |
| 11 | 4 | 256 | 295.24 | 9.38 | 31.48× | 32.00× |

Table 1: Tower trace optimization benchmarks. Here $\zeta$ is a primitive $2^m$-th root of unity, $k = 2^t$ with $t = 2$, and $H = \langle \sigma_{-1}, \sigma_{4k+1} \rangle$ with $|H| = 2^{m-1}/k$.

# References


[DF04] David S. Dummit and Richard M. Foote, *Abstract algebra*, 3rd ed., John Wiley & Sons, 2004.

[NOZ26] Ngoc Khanh Nguyen, George O'Rourke, and Jiapeng Zhang, *Hachi: Commitments from polynomial evaluation codes with smaller communication and faster verification*, 2026, Cryptology ePrint Archive, Report 2026/156.