The Code Equivalence Problem: New Algorithms and Reductions

Huck Bennett
University of Colorado Boulder

Based on joint works with Drisana Bhatia, Jean-François Biasse, Medha Durisheti, Lucas LaBuff, Kaung Myat Htay Win, Vincenzo Pallozzi Lavorante, and Philip Waitkevich.

UTSA ALGORITHMS SEMINAR, OCT. 10, 2025

This talk is based on two joint works

1. **Asymptotic improvements to provable algorithms for the code equivalence problem**
   with Drisana Bhatia, Jean-François Biasse, Medha Durisheti,
   Lucas LaBuff, Vincenzo Pallozzi Lavorante, and Philip Waitkevich
   (https://eprint.iacr.org/2025/187).
   In ISIT 2025 and accepted to IEEE Transactions on Information Theory.

2. **Relating Code Equivalence to Other Isomorphism Problems**
   with Kaung Myat Htay Win (https://eprint.iacr.org/2024/782).
   In Designs, Codes, and Cryptography 2025.

Computational Isomorphism Problems

Problem: Given two {graphs, codes, lattices} as input, decide if they are "essentially the same."

Graphs

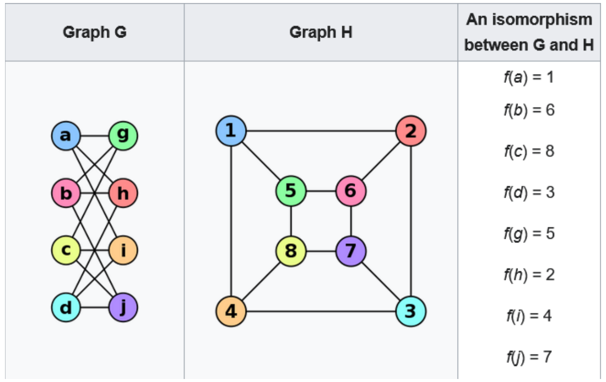
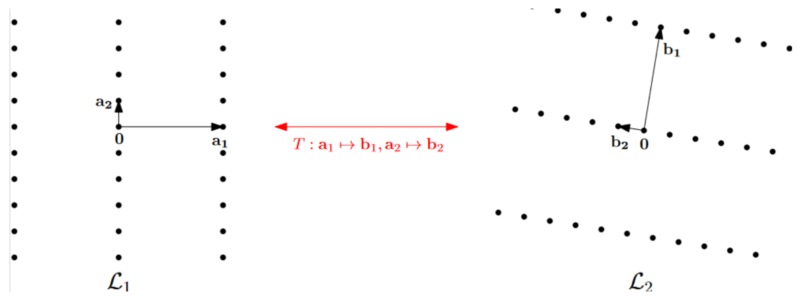


Image source: Wikipedia.

Codes

$$G = \begin{pmatrix} -1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \mapsto G' = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & -1 \end{pmatrix}$$

Lattices



Goals of our work:

1. Give faster algorithms for code equivalence.

2. Understand the relationship between code equivalence.
and isomorphism problems on graphs and lattices.

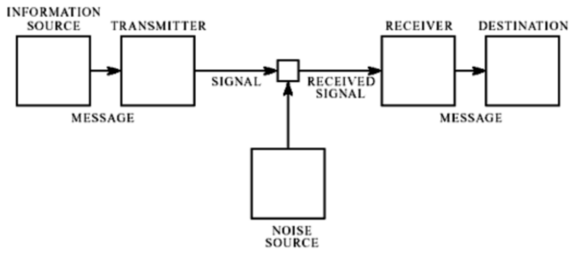
Cryptographic Motivation

Cryptography Based on Code Equivalence:

- McEliece Cryptosystem [McEliece, ’78].
- “Classic McEliece” NIST PQC Standardization Process Submission [Albrecht et al., ’22].
- LESS Identification Scheme [Biasse, Micheli, Persichetti, Santini, ’20].

Cryptography Based on Lattice Isomorphism:

- LIP-based KEM: (Ducas and van Woerden, ‘22).
- Rotations of $\mathbb{Z}^n$ PKC: [**Bennett**, Ganju, Peetathawatchai, and Stephens-Davidowitz, ’23].
- HAWK Digital Signature Scheme: [Ducas, Postlethwaite, Pulles, van Woerden, ’22].



From “A Mathematical Theory of Communication”

Coding Theory 101

Main use of error-correcting codes: Robust communication.

Want to encode a k-bit message in a redundant way.

Ex. 4-bit message m. Repeat each coordinate 3 times.
- $m \coloneqq (0,1,0,0) \mapsto c \coloneqq (0,0,0,1,1,1,0,0,0,0,0,0)$.
- c is $n = 12$ bits long, protects against 1 arbitrary error.

Ex. Hamming(7,4) code. Compute $C = Gm$, where $G \coloneqq \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$.
- $m \coloneqq (0,1,0,0) \mapsto c \coloneqq (1,0,0,1,1,0,0)$.
- c is $n = 7$ bits long, protects against 1 arbitrary error.

Want a code with n as *small* as possible and # of errors tolerated as *large* as possible.

Codes

Def. An $[n, k, d]_q$ code C is a linear subspace of $\mathbb{F}_q^n$ of dimension k with $\|x - y\|_0 \geq d$ for distinct $x, y \in C$.

- $\|\cdot\|_0$ denotes Hamming weight, the number of non-zero coordinates of a vector.
- $[n, k]_q$ denotes an $[n, k, d]_q$ code for some d.

Primal representation: Column basis generator matrix $G \in \mathbb{F}_q^{n \times k}$, $C(G) \coloneqq \left\{G\boldsymbol{z} : \boldsymbol{z} \in \mathbb{F}_q^k\right\}$.

Fact: $C(G_1) = C(G_2)$ if and only if $G_2 = G_1 U$ for an invertible matrix U (i.e., $U \in \mathrm{GL}_k(\mathbb{F}_q)$).

Code Equivalence Problem(s)

C_1, C_2 are *linearly equivalent* if there exists a monomial matrix M such that $MC_1 = C_2$.

- A monomial matrix M is such that $M = DP$ for full-rank diagonal D and permutation matrix P.

Ex. $$\begin{pmatrix} 0 & 0 & 3 \\ 1 & 0 & 0 \\ 0 & 4 & 0 \end{pmatrix} = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

C_1, C_2 are *permutationally equivalent* if there exists a permutation matrix P such that $PC_1 = C_2$.

Permutationally Equiv.:

$$G = \begin{pmatrix} 2 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}, G' = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 2 \end{pmatrix}$$

Linearly Perm., Not Perm. Equiv.:

$$G = \begin{pmatrix} 2 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}, G' = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Not Linearly Equiv.:

$$G = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}, G' = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}$$

Search Versions of Code Equivalence

Def. Linear Code Equivalence Problem (LCE) over $\mathbb{F}_q$: Given generator matrices G_1, G_2 of equivalent $[n, k]_q$ codes, find an $n \times n$ monomial matrix M and some $U \in \mathrm{GL}_k(\mathbb{F}_q)$ such that $MG_1U = G_2$ (if they exist).

Def. Permutation Code Equivalence Problem (PCE) over $\mathbb{F}_q$: Given generator matrices G_1, G_2 of equivalent $[n, k]_q$ codes, find an $n \times n$ permutation matrix P and some $U \in \mathrm{GL}_k(\mathbb{F}_q)$ such that $PG_1U = G_2$ (if they exist).

Def. Signed Permutation Code Equivalence (SPCE) over $\mathbb{F}_q$: Given generator matrices G_1, G_2 of $[n, k]$ codes, find an $n \times n$ signed permutation matrix S and some $U \in \mathrm{GL}_k\left(\mathbb{F}_q\right)$ such that $SG_1U = G_2$ (if they exist).

Graph Isomorphism Problem

Def. Graphs $G_1 = (V, E_1), G_2 = (V, E_2)$ are isomorphic if there exists a permutation $\pi : V \to V$ such that for all $u, v \in V$, $\{u, v\} \in E_1 \Leftrightarrow \{\pi(u), \pi(v)\} \in E_2$.

Graph Isomorphism Problem (GI): Decide if input graphs G_1, G_2 (represented by adjacency matrices) are isomorphic.

[Babai '16]: GI is solvable in $n^{\mathrm{poly}(\log n)}$ time.

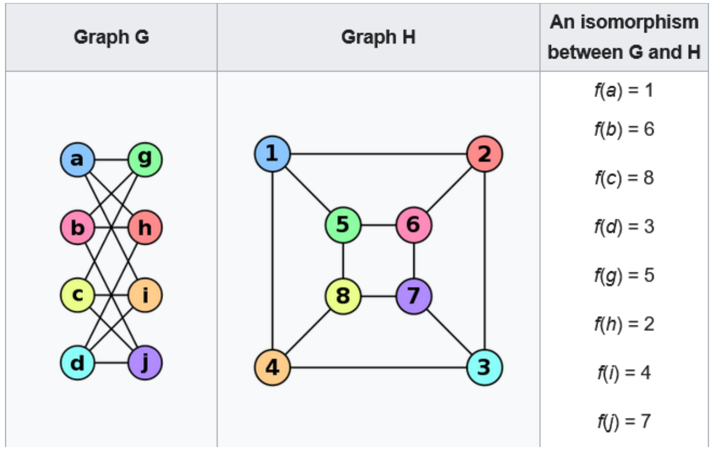


Image source: Wikipedia.

Part 1: Algorithms for Code Equivalence

Related Work

1. Heuristic algorithms using Information Set Decoding (ISD):
   a. [Leon ’82], [Beullens ’20], [Barenghi, Biasse, Persichetti and Santini ’23].
2. Algorithms for Codes with small hulls (the *hull* of C is $C \cap C^{\perp}$):
   a. Supporting splitting algorithm [Sendrier ’00].
   b. Reduction from [Bardet, Otmani and Saeed-Taha ’09].
3. Reduction from LCE on $[n, k]_q$ codes to PCE on $[(q-1)n, k]_q$ codes [Sendrier-Simos ’13].
4. **Provable general algorithms (our focus):**
   a. Deterministic PCE algorithm running in $2^{n+o(n)}$ time [Babai ‘11].
   b. Randomized LCE algorithm running in $2^{n/2+o(n)}$ time on *random* codes over fields of order $q \geq 7$ [Nowakowski ’25].

Question: What about *worst-case* codes and when $q < 7$?

Our Results

Theorem: The following algorithms exist for code equivalence over arbitrary $[n, k]_q$ codes for arbitrary prime powers q:

1. A $2^{n+o(n+q)}$-time deterministic algorithm for LCE.
2. A $2^{n/2+o(n+q)}$-time randomized algorithm for PCE and LCE.
3. A $2^{n/3+o(n+q)}$-time quantum algorithm for PCE and LCE.

Algorithm (1) complements the algorithm of [Babai ’11] for PCE.

Algorithms (2) and (3) resolve the question on the previous slide and remove both restrictions from algorithms with similar running times in [Nowakowski ’25].

High-level idea for (2) and (3): Reduce code equivalence to collision/claw finding.

Babai's PCE Algorithm

Let C be an $[n,k]_q$ code and let $G \in \mathbb{F}_q^{n\times k}$ be a generator matrix of C.

Def. An *information set* C is a set $S = \{i_1, \dots, i_k\} \subseteq [n]$ of coordinates such that the $k \times k$ matrix obtained by restricting G to rows indexed by S is full-rank.

Def. If $G = \begin{pmatrix} I_k \\ G' \end{pmatrix}$ for $G' \in \mathbb{F}_q^{(n-k)\times k}$ then it is said to be in *systematic form.*

- If $[k] \coloneqq \{1, \dots, k\}$ is an information set, then it is easy to put $G = \begin{pmatrix} G' \\ G'' \end{pmatrix}$ into systematic form by setting

$$G \coloneqq \begin{pmatrix} G' \\ G'' \end{pmatrix} \cdot (G')^{-1} = \begin{pmatrix} I_k \\ G'' \cdot (G')^{-1} \end{pmatrix}.$$

Def. Information sets J_1, J_2 of equivalent codes C_1, C_2 respectively are called *matching* if there exists a permutation $\pi: [n] \to [n]$ such that $\pi(C_1) = C_2$ and $\pi(J_1) = J_2$.

- **Observation:** A permutation such that $\pi(C_1) = C_2$ must map information sets to information sets.

Main idea of algorithm: To solve PCE, it suffices to know a pair of matching information sets and to make one call to a graph isomorphism (GI) oracle.

Babai's PCE Algorithm

Let G_1, G_2 be generator matrices of equivalent codes C_1, C_2 with matching information sets J_1, J_2.

Assume WLOG that $J_1 = J_2 = \{1, \ldots, k\}$. If not, permute coordinates of C_1, C_2 so that this holds.

Put G_1, G_2 into systematic form so that $G_1 = \begin{pmatrix} I_k \\ G_1' \end{pmatrix}, G_2 = \begin{pmatrix} I_k \\ G_2' \end{pmatrix}$.

Then by the assumption that J_1, J_2 are matching, there must exist $P_1 \in \mathcal{P}_k, P_2 \in \mathcal{P}_{n-k}$,
$U \in \mathrm{GL}_k(\mathbb{F}_q)$ such that

$$\begin{pmatrix} P_1^{-1} & 0 \\ 0 & P_2 \end{pmatrix} \begin{pmatrix} I_k \\ G_1' \end{pmatrix} U = \begin{pmatrix} I_k \\ G_2' \end{pmatrix}.$$

This implies that $U = P_1$ and therefore $P_2 G_1' P_1 = G_2'$.

Babai's PCE Algorithm

To recover a permutation from C_1 to C_2, it suffices to find permutation matrices P_1, P_2 such that $P_2 G_1' P_1 = G_2'$.

Observation [Babai 2011]: This is equivalent to graph isomorphism on $(n-k, k)$-bipartite graphs with $\mathbb{F}_q$-labeled edges!

- Regard G_1', G_2' as adjacency matrices of such graphs.

Theorem [Babai 2016]: There is a quasipolynomial-time algorithm for GI.

Babai's PCE algorithm:

- Compute an arbitrary information set J_1 of C_1.
- Enumerate all $\binom{n}{k} \leq 2^n$ size-k subsets of indices J_2 corresponding to candidate information sets of C_2 matching J_1.
- Solve the resulting GI instance in $2^{o(n)}$ time.

Takes $2^{n+o(n)}$ time.

Our Randomized $2^{n/2}$-time Algorithm for PCE

Theorem [Babai 2019]: There is a quasipolynomial-time computable *canonical form* for graphs.

- A canonical form $F: \mathbb{F}_q^{m\times n} \to \mathbb{F}_q^{m\times n}$ for $\mathbb{F}_q$-edge labeled bipartite graphs is a function such that:
  (1) $F(A) = A$,
  (2) $F(A_1) = F(A_2)$ if and only if A_1 and A_2 are adjacency matrices of isomorphic graphs.

Let f_i for $i = 1, 2$ be a function mapping information sets of C_i to $F\left(G_i'\right)$, where $G_i = \begin{pmatrix} I_k \\ G_i' \end{pmatrix}$.

- Interpret G_i' as the adjacency matrix of a bipartite graph.
- Note that f_1, f_2 have the same range.

Key idea: We have reduced PCE to finding a pair (J_1, J_2) of information sets of C_1, C_2 such that $f_1(J_1) = f_2(J_2)$.

- Such a pair (J_1, J_2) is called a *claw.*

Each code C_i has the same number $N \leq \binom{n}{k} \leq 2^n$ of information sets.

Our Randomized $2^{n/2}$-time Algorithm for PCE

Sample m independent, uniformly random information sets from each of C_1, C_2.

- I.e., sample information sets $J_1, \ldots, J_m$ of C_1 and $J'_1, \ldots, J'_m$ of C_2.
- The expected number of matching information sets/claws (J_i, J'_j) is at least m^2/N.

So, setting $m \approx N^{1/2}$, we get that the expected number of claws is at least 1.

- Runs in roughly $N^{1/2} \leq 2^{n/2}$ time.

Issue #1: Showing that this works with high probability.

Solution: Bound variance of expected number of claws, apply Chebyshev's inequality to show that it concentrates around expectation.

Issue #2: How do we sample uniformly random information sets?

Solution: We don't. Instead, we use an algorithm for matroid basis sampling [Anari, Liu, Oveis Gharan, Vinzant '19] that efficiently samples *nearly* uniformly random bases efficiently.

Extension: We also extend this to a $2^{n/3}$-time quantum algorithm for PCE.

Thank you!