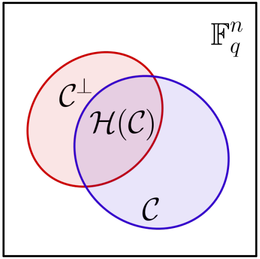
The Mysterious Case of Code Equivalence

Violetta Weger

VT-Swiss Summer School

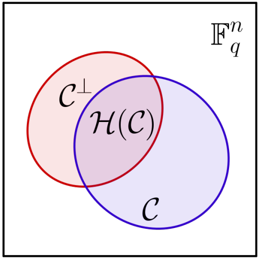
July 1-5, 2024

Basics



- Linear code: $\mathcal{C} \subseteq \mathbb{F}_q^n$ linear subspace of dimension k
- Generator matrix: $G \in \mathbb{F}_q^{k \times n}$ with $\langle G \rangle = \mathcal{C}$
- Dual code: $\mathcal{C}^\perp = \{x \in \mathbb{F}_q^n \mid \langle x, c \rangle = 0 \ \forall \ c \in \mathcal{C}\}$
- Parity-check matrix: $H \in \mathbb{F}_q^{n-k \times n}$ with $\langle H \rangle = \mathcal{C}^\perp$
- Hull: $\mathcal{H}(\mathcal{C}) = \mathcal{C} \cap \mathcal{C}^\perp$

Basics

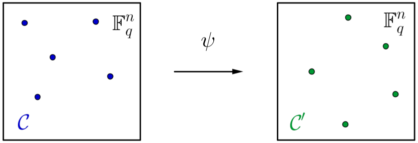


$$\begin{aligned} cG^\top &= mGG^\top = 0 \\ \dim(\ker(GG^\top)) &= k - \mathrm{rk}(GG^\top) \\ &= 0 \text{ w.h.p} \end{aligned}$$

- Linear code: $\mathcal{C} \subseteq \mathbb{F}_q^n$ linear subspace of dimension k
- Generator matrix: $G \in \mathbb{F}_q^{k \times n}$ with $\langle G \rangle = \mathcal{C}$
- Dual code: $\mathcal{C}^\perp = \{x \in \mathbb{F}_q^n \mid \langle x, c \rangle = 0 \ \forall \ c \in \mathcal{C}\}$
- Parity-check matrix: $H \in \mathbb{F}_q^{n-k \times n}$ with $\langle H \rangle = \mathcal{C}^\perp$
- Hull: $\mathcal{H}(\mathcal{C}) = \mathcal{C} \cap \mathcal{C}^\perp = \{0\}$ w.h.p.

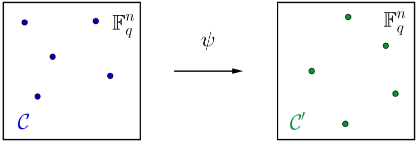
Basics

- Hamming weight: $\mathrm{wt}(c) = |\{i \in \{1, \ldots, n\} \mid c_i \neq 0\}|$
- Linear isometry: linear map $\psi : \mathbb{F}_q^n \to \mathbb{F}_q^n$ with $\mathrm{wt}(c) = \mathrm{wt}(\psi(c))\ \forall c \in \mathbb{F}_q^n$
- Hamming isometries $\mathcal{L} = (\mathbb{F}_q^\star)^n \rtimes (\mathrm{Aut}(\mathbb{F}_q) \times \mathcal{S}_n)$
- Code equivalence $\mathcal{C}$ is equivalent to $\mathcal{C}'$ if exists $\psi \in \mathcal{L} : \psi(\mathcal{C}) = \mathcal{C}'$



Basics

- Hamming weight: $\mathrm{wt}(c) = |\{i \in \{1, \ldots, n\} \mid c_i \neq 0\}|$
- Linear isometry: linear map $\psi : \mathbb{F}_q^n \to \mathbb{F}_q^n$ with $\mathrm{wt}(c) = \mathrm{wt}(\psi(c))\ \forall c \in \mathbb{F}_q^n$
- Hamming isometries $\mathcal{L} = (\mathbb{F}_q^\star)^n \rtimes (\mathrm{Aut}(\mathbb{F}_q) \times \mathcal{S}_n)$
- Code equivalence $\mathcal{C}$ is equivalent to $\mathcal{C}'$ if exists $\psi \in \mathcal{L} : \psi(\mathcal{C}) = \mathcal{C}'$



Coding Theory: Distinguish if codes belong to new class or not

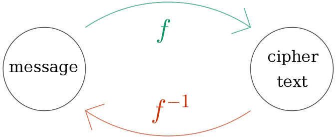
E. M. Gabidulin, "New Rank Codes with Efficient Decoding", EnT, 2017.

A. Neri, S. Puchinger, A.-L. Horlemann, "Invariants and Inequivalence of Linear Rank-Metric Codes.", ISIT, 2019.

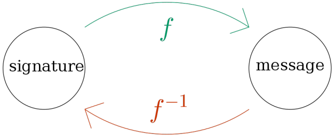
Motivation

Public-key cryptography

Encryption



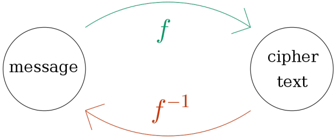
Signature



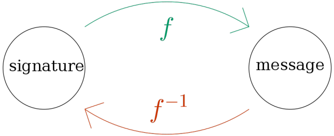
Motivation

Public-key cryptography

Encryption



Signature

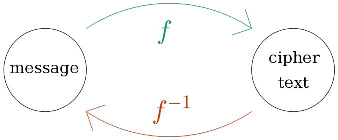


- f easy to compute with
- f^{-1} hard to compute with
- f^{-1} easy with secret

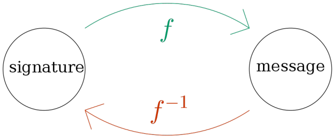
Motivation

Public-key cryptography

Encryption



Signature



computing f^{-1}: hard mathematical problem

Basics

- Code equivalence: $\mathcal{C}$ is equivalent to $\mathcal{C}'$ if exists $\psi \in \mathcal{L} : \psi(\mathcal{C}) = \mathcal{C}'$
- Linear equivalence: $\mathcal{C}$ is linear equivalent to $\mathcal{C}'$ if $\exists \varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n : \varphi(\mathcal{C}) = \mathcal{C}'$
- Permutation equivalence: $\mathcal{C}$ is permutation equivalent to $\mathcal{C}'$ if $\exists \sigma \in \mathcal{S}_n$: $\sigma(\mathcal{C}) = \mathcal{C}'$

Basics

- Code equivalence: $\mathcal{C}$ is equivalent to $\mathcal{C}'$ if exists $\psi \in \mathcal{L} : \psi(\mathcal{C}) = \mathcal{C}'$
- Linear equivalence: $\mathcal{C}$ is linear equivalent to $\mathcal{C}'$ if $\exists \varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n : \varphi(\mathcal{C}) = \mathcal{C}'$
- Permutation equivalence: $\mathcal{C}$ is permutation equivalent to $\mathcal{C}'$ if $\exists \sigma \in \mathcal{S}_n$: $\sigma(\mathcal{C}) = \mathcal{C}'$

Linear Equivalence Problem (LEP):
Given $\mathcal{C}, \mathcal{C}'$ find $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$: $\varphi(\mathcal{C}) = \mathcal{C}'$

Permutation Equivalence Problem (PEP):
Given $\mathcal{C}, \mathcal{C}'$ find $\sigma \in \mathcal{S}_n$: $\sigma(\mathcal{C}) = \mathcal{C}'$

Basics

- Code equivalence: $\mathcal{C}$ is equivalent to $\mathcal{C}'$ if exists $\psi \in \mathcal{L} : \psi(\mathcal{C}) = \mathcal{C}'$
- Linear equivalence: $\mathcal{C}$ is linear equivalent to $\mathcal{C}'$ if $\exists \varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n : \varphi(\mathcal{C}) = \mathcal{C}'$
- Permutation equivalence: $\mathcal{C}$ is permutation equivalent to $\mathcal{C}'$ if $\exists \sigma \in \mathcal{S}_n$: $\sigma(\mathcal{C}) = \mathcal{C}'$

Linear Equivalence Problem (LEP):
Given G, G' find $S \in \mathrm{GL}_k(q), P \in S_n, D = \mathrm{diag}(v) : SGPD = G'$

Permutation Equivalence Problem (PEP):
Given G, G' find $S \in \mathrm{GL}_k(q), P \in S_n : SGP = G'$

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover

secret

Verifier

public

✓

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
	$\xrightarrow{\text{commitment}}$
	$\xleftarrow{\text{challenge}}$
secret	$\xrightarrow{\text{response}}$	public
		✓

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover	Interaction	Verifier
	$\xrightarrow{\text{commitment}}$	
	$\xleftarrow{\text{challenge}}$	
secret	$\xrightarrow{\text{response}}$	public
		✓

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
secret	$\xrightarrow{\text{Fiat-Shamir}}$	public
		✓

$\rightarrow$ Signature scheme

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
	$\xrightarrow{\text{commitment}}$	
	$\xleftarrow{\text{challenge}}$	
secret	$\xrightarrow{\text{response}}$	public
		✓

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover

secret

$$\xrightarrow{\text{commitment}}$$
$$\xleftarrow{\text{challenge}}$$
$$\xrightarrow{\text{response}}$$

Verifier

public

✓

$$G \xrightarrow{\varphi} G'$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
	$\xrightarrow{\text{commitment}}$	
	$\xleftarrow{\text{challenge}}$	
secret	$\xrightarrow{\text{response}}$	public
		✓

$$\begin{tikzcd} G \arrow[rr, "\varphi"] \arrow[dr, "\widetilde{\varphi}"'] & & G' \\ & \widetilde{G} & \end{tikzcd}$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Commitment: $\widetilde{G}$ such that $\widetilde{\varphi}(G) = \widetilde{G}$

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover — **Verifier**

$\xrightarrow{\text{commitment}}$

$\xleftarrow{\text{challenge}}$

$\xrightarrow{\text{response}}$

secret — public

✓

$$\begin{tikzcd}
G \arrow[rr, "\varphi"] \arrow[dr, "\widetilde{\varphi}"'] & & G' \arrow[dl, "\bar{\varphi}"] \\
& \widetilde{G} &
\end{tikzcd}$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Commitment: $\widetilde{G}$ such that $\widetilde{\varphi}(G) = \widetilde{G}$

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
	$\xrightarrow{\text{commitment}}$	
	$\xleftarrow{\text{challenge}}$	
secret	$\xrightarrow{\text{response}}$	public
		✓

$$\begin{tikzcd} G \arrow[rr, "\varphi"] \arrow[dr, "\widetilde{\varphi}"'] & & G' \arrow[dl, "\bar{\varphi} = \varphi^{-1} \circ \widetilde{\varphi}"] \\ & \widetilde{G} & \end{tikzcd}$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Commitment: $\widetilde{G}$ such that $\widetilde{\varphi}(G) = \widetilde{G}$

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover | **Verifier**

$\xrightarrow{\text{commitment}}$

$\xleftarrow{\text{challenge}}$

$\xrightarrow{\text{response}}$

secret | public

✓

$$\begin{tikzcd}
G \arrow[rr, "\varphi"] \arrow[dr, "\widetilde{\varphi}"'] & & G' \arrow[dl, "\bar{\varphi}"] \\
& \widetilde{G} &
\end{tikzcd}$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Commitment: $\widetilde{G}$ such that $\widetilde{\varphi}(G) = \widetilde{G}$

Challenges: 1. show $\widetilde{\varphi}$, 2. show $\bar{\varphi}$

ZK: $\widetilde{\varphi}$, $\bar{\varphi}$ do not reveal anything on φ

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
	$\xrightarrow{\text{commitment}}$	
	$\xleftarrow{\text{challenge}}$	
secret	$\xrightarrow{\text{response}}$	public
		✓

$$
\begin{tikzcd}
G \arrow[rr, "?"] \arrow[dr, "\widetilde{\varphi}"'] & & G' \\
& \widetilde{G} &
\end{tikzcd}
$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Commitment: $\widetilde{G}$ such that $\widetilde{\varphi}(G) = \widetilde{G}$

Challenges: 1. show $\widetilde{\varphi}$, 2. show $\bar{\varphi}$

ZK: $\widetilde{\varphi}$, $\bar{\varphi}$ do not reveal anything on φ

Soundness error: t rounds 2^{-t}

Motivation

Can build Zero-Knowledge (ZK) protocol from group action

Prover		Verifier
secret	$\xrightarrow{\text{commitment}}$ $\xleftarrow{\text{challenge}}$ $\xrightarrow{\text{response}}$	public
	✓

$$
\begin{tikzcd}
G \arrow[rr, "?"] \arrow[dr, "\widetilde{\varphi}"'] & & G' \arrow[dl, "?"] \\
& \widetilde{G} &
\end{tikzcd}
$$

secret key: $\varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n$

public key: G, G' with $\varphi(G) = G'$

Commitment: $\widetilde{G}$ such that $\widetilde{\varphi}(G) = \widetilde{G}$

Challenges: 1. show $\widetilde{\varphi}$, 2. show $\bar{\varphi}$

ZK: $\widetilde{\varphi}$, $\bar{\varphi}$ do not reveal anything on φ

Soundness error: t rounds 2^{-t}

Motivation

ongoing NIST standardization process for post-quantum signature schemes

- LESS linear equivalence
- MEDS matrix code equivalence
- PERK subcode equivalence

Main question: How hard is code equivalence?

- complexity class
- solvers

Motivation

ongoing NIST standardization process for post-quantum signature schemes

- LESS   linear equivalence
- MEDS   matrix code equivalence
- PERK   subcode equivalence

Main question: How hard is code equivalence?

- complexity class
- solvers

Motivation

ongoing NIST standardization process for post-quantum signature schemes

- LESS linear equivalence
- MEDS matrix code equivalence
- PERK subcode equivalence

Main question: How hard is code equivalence?

- complexity class
- can reduce PEP to GI

- solvers
- can reduce LEP to PEP if $q < 5$

Is Code Equivalence NP-hard?

Is Code Equivalence NP-hard?

...no

Is Code Equivalence NP-hard?

...no

Arthur-Merlin Protocol

Merlin has a ”no” instance: $\mathcal{C}_1, \mathcal{C}_2$, wants to convince Arthur $\nexists\varphi$

- Arthur chooses G_i and ψ
- Arthur sends $G' = \psi(G_i)$
- Merlin replies with i

- t rounds $\rightarrow 2^{-t}$
- not NP-hard, else AM = PH $\rightarrow$ complexity hierarchy collapses

Sneak Peek

Reduction from PEP to GI

$\mathcal{G} = (V, E)$ weighted graph weight on edge $\{u, v\}$ is $w(u, v)$ $V = [1, n]$

Graph Isomorphism (GI)

Given $\mathcal{G} = (V, E), \mathcal{G}' = (V, E')$, find $\sigma \in \mathcal{S}_n$, s.t.

1. $\{u, v\} \in E \leftrightarrow \{\sigma(u), \sigma(v)\} \in E'$ 2. $w(u, v) = w(\sigma(u), \sigma(v))$

Sneak Peek

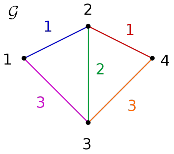
Reduction from PEP to GI

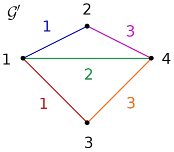
$\mathcal{G} = (V, E)$ weighted graph   weight on edge $\{u, v\}$ is $w(u, v)$ $V = [1, n]$

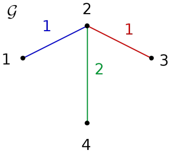
Graph Isomorphism (GI)

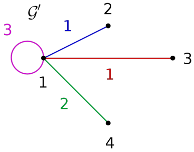
Given $\mathcal{G} = (V, E), \mathcal{G}' = (V, E')$, find $\sigma \in \mathcal{S}_n$, s.t.

1. $\{u, v\} \in E \leftrightarrow \{\sigma(u), \sigma(v)\} \in E'$ 2. $w(u, v) = w(\sigma(u), \sigma(v))$









Sneak Peek

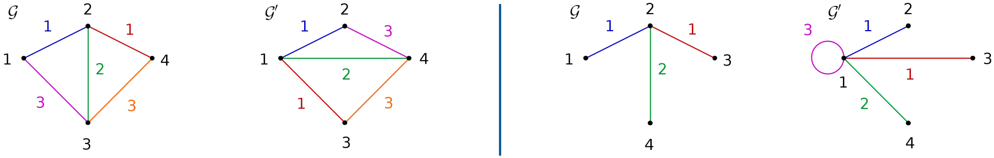
Reduction from PEP to GI

$\mathcal{G} = (V, E)$ weighted graph weight on edge $\{u, v\}$ is $w(u, v)$ $V = [1, n]$

Graph Isomorphism (GI)

Given $\mathcal{G} = (V, E), \mathcal{G}' = (V, E')$, find $\sigma \in \mathcal{S}_n$, s.t.

1. $\{u, v\} \in E \leftrightarrow \{\sigma(u), \sigma(v)\} \in E'$ 2. $w(u, v) = w(\sigma(u), \sigma(v))$



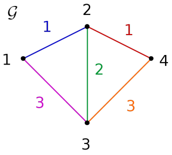
L. Babai. "Graph isomorphism in quasipolynomial time", ACM, 2016.

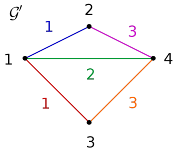
Sneak Peek

Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric

Sneak Peek

Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric

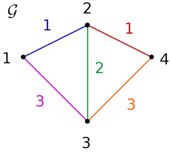


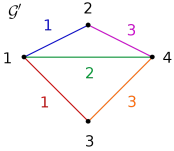


$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

Sneak Peek

Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric



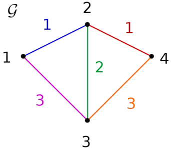


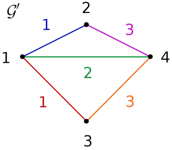
$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

GI: $\sigma(\mathcal{G}) = \mathcal{G}' \leftrightarrow P^\top A P = A'$

Sneak Peek

Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric



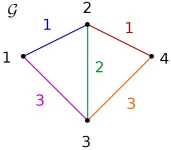


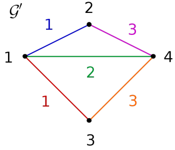
$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

GI: $\sigma(\mathcal{G}) = \mathcal{G}' \leftrightarrow P^\top A P = A'$ PEP: $SAP = A'$

Sneak Peek

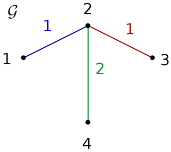
Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric

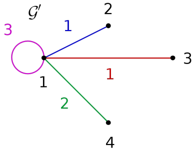




$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

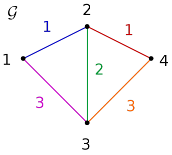
GI: $\sigma(\mathcal{G}) = \mathcal{G}' \leftrightarrow P^\top A P = A'$ PEP: $SAP = A'$

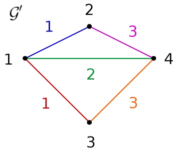




Sneak Peek

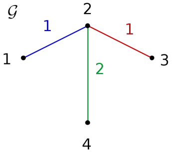
Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric

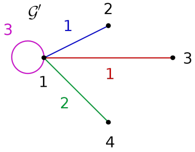




$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

$$\text{GI:} \quad \sigma(\mathcal{G}) = \mathcal{G}' \leftrightarrow P^\top A P = A' \qquad \text{PEP:} \quad S A P = A'$$

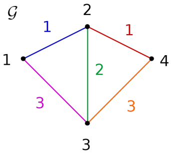


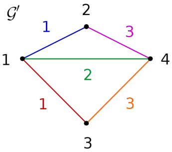


$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 3 & 1 & 1 & 2 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{pmatrix}$$

Sneak Peek

Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\rightarrow$ symmetric





$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

$$\text{GI:} \quad \sigma(\mathcal{G}) = \mathcal{G}' \leftrightarrow P^\top A P = A' \qquad \text{PEP:} \quad SAP = A'$$

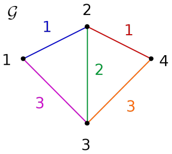
Reduction PEP $\rightarrow$ GI:

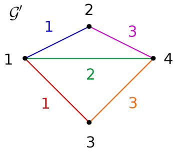
instance $\mathcal{C}, \mathcal{C}'$ $\rightarrow$ solve instance $\mathcal{G}, \mathcal{G}'$ $\rightarrow$ solution for PEP

$\rightarrow$ PEP easier than GI (quasi-polynomial)

Sneak Peek

Adjacency matrix A: $A_{i,j} = w(i,j)$ if $\{i,j\} \in E$ and 0 else. $\quad \rightarrow$ symmetric





$$A = \begin{pmatrix} 0 & 1 & 3 & 0 \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix} \qquad A' = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 3 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

GI: $\quad \sigma(\mathcal{G}) = \mathcal{G}' \leftrightarrow P^\top A P = A'$ $\qquad$ PEP: $\quad SAP = A'$

Reduction PEP $\rightarrow$ GI:

instance $\mathcal{C}, \mathcal{C}'$ $\quad \rightarrow \quad$ solve instance $\mathcal{G}, \mathcal{G}'$ $\quad \rightarrow \quad$ solution for PEP

$\rightarrow$ PEP easier than GI (quasi-polynomial)

How to choose A for code $\mathcal{C}$ to form graph $\mathcal{G}$?

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull:

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull:

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull:

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

$\langle A \rangle = \mathcal{C}$ A independent of G A symmetric

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull:

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

$\langle A \rangle = \mathcal{C}$ | A independent of G | A symmetric

$$(SG)^\top (SG(SG)^\top)^{-1} SG = G^\top (GG^\top)^{-1} G$$

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull:

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

$\langle A \rangle = \mathcal{C}$ $\qquad$ A independent of G $\qquad$ A symmetric

take $\mathcal{G}, \mathcal{G}'$ having adjacency matrices A, A'

$$\sigma(\mathcal{C}) = \mathcal{C}' \leftrightarrow \sigma(\mathcal{G}) = \mathcal{G}'$$

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull:

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

$\langle A \rangle = \mathcal{C}$ $\quad$ A independent of G $\quad$ A symmetric

take $\mathcal{G}, \mathcal{G}'$ having adjacency matrices A, A'

$$\sigma(\mathcal{C}) = \mathcal{C}' \leftrightarrow \sigma(\mathcal{G}) = \mathcal{G}'$$

$\leftarrow$: If $\sigma(\mathcal{G}) = \mathcal{G}'$ then $P^\top AP = A'$ $\quad \rightarrow \sigma(\mathcal{C}) = \mathcal{C}'$

Sneak Peek

M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.

For $\mathcal{C} = \langle G \rangle$ with trivial hull: only works for instance $(\mathcal{C}, \mathcal{C}')$ w.h.p.

$$A = G^\top (GG^\top)^{-1} G \in \mathbb{F}_q^{n \times n}$$

$\langle A \rangle = \mathcal{C}$ A independent of G A symmetric

take $\mathcal{G}, \mathcal{G}'$ having adjacency matrices A, A'

$$\sigma(\mathcal{C}) = \mathcal{C}' \leftrightarrow \sigma(\mathcal{G}) = \mathcal{G}'$$

$\rightarrow$: If $\sigma(\mathcal{C}) = \mathcal{C}'$ then for any gen. matrices G, G' ex. S: $SGP = G'$

$$A' = (GP)^\top (GP(GP)^\top)^{-1} GP = P^\top G^\top (GG^\top)^{-1} GP = P^\top AP$$

Sneak Peek

Reduction from LEP to PEP and why only for $q < 5$

Closure of code: For $\mathcal{C} \subset \mathbb{F}_q^n$ its closure is

$$\widetilde{\mathcal{C}} = \{(\alpha c_i)_{(i,\alpha)\in[1,n]\times\mathbb{F}_q^\star} \mid (c_i)_{i\in[1,n]} \in \mathcal{C}\} \subset \mathbb{F}_q^{n(q-1)}$$

Sneak Peek

Reduction from LEP to PEP and why only for $q < 5$

Closure of code: For $\mathcal{C} \subset \mathbb{F}_q^n$ its closure is

$$\widetilde{\mathcal{C}} = \{(\alpha c_i)_{(i,\alpha)\in[1,n]\times\mathbb{F}_q^\star} \mid (c_i)_{i\in[1,n]} \in \mathcal{C}\} \subset \mathbb{F}_q^{n(q-1)}$$

$$G = \begin{pmatrix} | & & | \\ g_1 & \cdots & g_n \\ | & & | \end{pmatrix} \to \widetilde{G} = \begin{pmatrix} | & | & & | & & | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_1 & \cdots & g_n & \alpha g_n & \cdots & \alpha^{q-2} g_n \\ | & | & & | & & | & | & & | \end{pmatrix}$$

Sneak Peek

Reduction from LEP to PEP and why only for $q < 5$

Closure of code: For $\mathcal{C} \subset \mathbb{F}_q^n$ its closure is

$$\widetilde{\mathcal{C}} = \{(\alpha c_i)_{(i,\alpha)\in[1,n]\times\mathbb{F}_q^\star} \mid (c_i)_{i\in[1,n]} \in \mathcal{C}\} \subset \mathbb{F}_q^{n(q-1)}$$

$$G = \begin{pmatrix} | & & | \\ g_1 & \cdots & g_n \\ | & & | \end{pmatrix} \to \widetilde{G} = \begin{pmatrix} | & | & & | & & | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_1 & \cdots & g_n & \alpha g_n & \cdots & \alpha^{q-2} g_n \\ | & | & & | & & | & | & & | \end{pmatrix}$$

$$\exists \varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n : \varphi(\mathcal{C}) = \mathcal{C}' \quad \leftrightarrow \quad \exists \sigma \in \mathcal{S}_n : \sigma(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}'$$

Sneak Peek

Reduction from LEP to PEP and why only for $q < 5$

Closure of code: For $\mathcal{C} \subset \mathbb{F}_q^n$ its closure is

$$\widetilde{\mathcal{C}} = \{(\alpha c_i)_{(i,\alpha)\in[1,n]\times\mathbb{F}_q^\star} \mid (c_i)_{i\in[1,n]} \in \mathcal{C}\} \subset \mathbb{F}_q^{n(q-1)}$$

$$G = \begin{pmatrix} | & & | \\ g_1 & \cdots & g_n \\ | & & | \end{pmatrix} \to \widetilde{G} = \begin{pmatrix} | & | & & | & & | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_1 & \cdots & g_n & \alpha g_n & \cdots & \alpha^{q-2} g_n \\ | & | & & | & & | & | & & | \end{pmatrix}$$

$$\exists \varphi \in (\mathbb{F}_q^\star)^n \rtimes \mathcal{S}_n : \varphi(\mathcal{C}) = \mathcal{C}' \quad \leftrightarrow \quad \exists \sigma \in \mathcal{S}_n : \sigma(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}'$$

$$G P \mathrm{diag}(v) = G' \quad \leftrightarrow \quad \widetilde{G}\widetilde{P} = \widetilde{G}'$$

$$\widetilde{P} = \begin{pmatrix} P_1 & & \\ & \ddots & \\ & & P_n \end{pmatrix} Q, \; P_i \in S_{q-1}, \; Q \text{ block permutation}$$

Sneak Peek

.. and why only for $q < 5$?

Sneak Peek

.. and why only for $q < 5$?

For $q \geq 5$ $\quad \widetilde{\mathcal{C}} \subset \widetilde{\mathcal{C}}^{\perp}$ weakly self dual and $\mathcal{H}(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}$ $\lightning$

Sneak Peek

.. and why only for $q < 5$?

For $q \geq 5$ $\quad \widetilde{\mathcal{C}} \subset \widetilde{\mathcal{C}}^{\perp}$ weakly self dual and $\mathcal{H}(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}$ ⚡

$$X = \widetilde{G}\widetilde{G}^{\top} = \begin{pmatrix} | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} - & g_1 & - \\ - & \alpha g_1 & - \\ & \vdots & \\ - & \alpha^{q-2} g_n & - \end{pmatrix}$$

$$X_{i,j} = \sum_{\ell=1}^{n} g_{\ell,i} g_{\ell,j} \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2$$

Sneak Peek

.. and why only for $q < 5$?

For $q \geq 5$ $\quad \widetilde{\mathcal{C}} \subset \widetilde{\mathcal{C}}^{\perp}$ weakly self dual and $\mathcal{H}(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}$ ↯

$$X = \widetilde{G}\widetilde{G}^{\top} = \begin{pmatrix} | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} - & g_1 & - \\ - & \alpha g_1 & - \\ & \vdots & \\ - & \alpha^{q-2} g_n & - \end{pmatrix}$$

$$X_{i,j} = \sum_{\ell=1}^{n} g_{\ell,i} g_{\ell,j} \sum_{\beta \in \mathbb{F}_q^\star} \beta^2$$

Sneak Peek

.. and why only for $q < 5$?

For $q \geq 5$ $\quad \widetilde{\mathcal{C}} \subset \widetilde{\mathcal{C}}^{\perp}$ weakly self dual and $\mathcal{H}(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}$ $\lightning$

$$X = \widetilde{G}\widetilde{G}^{\top} = \begin{pmatrix} | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} - & g_1 & - \\ - & \alpha g_1 & - \\ & \vdots & \\ - & \alpha^{q-2} g_n & - \end{pmatrix}$$

$$X_{i,j} = \sum_{\ell=1}^{n} g_{\ell,i} g_{\ell,j} \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2$$

$\alpha^2 \neq 1$

$$\sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2 = \sum_{\beta \in \mathbb{F}_q^{\star}} (\alpha\beta)^2 = \alpha^2 \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2 \quad \rightarrow \quad \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2 = 0$$

Sneak Peek

.. and why only for $q < 5$?

For $q \geq 5$ $\quad \widetilde{\mathcal{C}} \subset \widetilde{\mathcal{C}}^{\perp}$ weakly self dual and $\mathcal{H}(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}$ ⚡

$$X = \widetilde{G}\widetilde{G}^{\top} = \begin{pmatrix} | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} - & g_1 & - \\ - & \alpha g_1 & - \\ & \vdots & \\ - & \alpha^{q-2} g_n & - \end{pmatrix}$$

$$X_{i,j} = \sum_{\ell=1}^{n} g_{\ell,i} g_{\ell,j} \sum_{\beta \in \mathbb{F}_q^\star} \beta^2 = 0$$

$\alpha^2 \neq 1$

$$\sum_{\beta \in \mathbb{F}_q^\star} \beta^2 = \sum_{\beta \in \mathbb{F}_q^\star} (\alpha\beta)^2 = \alpha^2 \sum_{\beta \in \mathbb{F}_q^\star} \beta^2 \quad \rightarrow \quad \sum_{\beta \in \mathbb{F}_q^\star} \beta^2 = 0$$

Sneak Peek

.. and why only for $q < 5$?

For $q \geq 5$ $\quad \widetilde{\mathcal{C}} \subset \widetilde{\mathcal{C}}^{\perp}$ weakly self dual and $\mathcal{H}(\widetilde{\mathcal{C}}) = \widetilde{\mathcal{C}}$ ↯

$$X = \widetilde{G}\widetilde{G}^{\top} = \begin{pmatrix} | & | & & | \\ g_1 & \alpha g_1 & \cdots & \alpha^{q-2} g_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} - & g_1 & - \\ - & \alpha g_1 & - \\ & \vdots & \\ - & \alpha^{q-2} g_n & - \end{pmatrix}$$

$$X_{i,j} = \sum_{\ell=1}^{n} g_{\ell,i} g_{\ell,j} \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2$$

$\alpha^2 \neq 1$ $\quad$ needs $q \geq 4$

$$\sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2 = \sum_{\beta \in \mathbb{F}_q^{\star}} (\alpha\beta)^2 = \alpha^2 \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2 \quad \rightarrow \quad \sum_{\beta \in \mathbb{F}_q^{\star}} \beta^2 = 0$$

What is known/Spoilers

Complexity Class

- Code equivalence is not NP-hard
- Easy instance of PEP:
  Random codes!

- rand. reduction from PEP to GI
- reduction from LEP to PEP
  if $q \geq 5$: weakly self dual

What is known/Spoilers

Complexity Class

- Code equivalence is not NP-hard
- Easy instance of PEP:
  Random codes!

- rand. reduction from PEP to GI
- reduction from LEP to PEP
  if $q \geq 5$: weakly self dual

Solvers

$$\begin{tikzcd} \mathcal{C} \arrow[d, "\varphi"'] \\ \mathcal{C}' \end{tikzcd}$$

What is known/Spoilers

Complexity Class

- Code equivalence is not NP-hard
- Easy instance of PEP:
  Random codes!

- rand. reduction from PEP to GI
- reduction from LEP to PEP
  if $q \geq 5$: weakly self dual

Solvers

$$
\begin{tikzcd}
\mathcal{C} \arrow[r] \arrow[d, "\varphi"'] & \mathcal{D} \\
\mathcal{C}' \arrow[r] & \mathcal{D}'
\end{tikzcd}
$$

What is known/Spoilers

Complexity Class

- Code equivalence is not NP-hard
- Easy instance of PEP:
  Random codes!

- rand. reduction from PEP to GI
- reduction from LEP to PEP
  if $q \geq 5$: weakly self dual

Solvers

$$\begin{tikzcd}
\mathcal{C} \arrow[r] \arrow[d, "\varphi"'] & \mathcal{D} & A_w(\mathcal{D}) \arrow[d, leftrightarrow, "= ?"] \\
\mathcal{C}' \arrow[r] & \mathcal{D}' & A_w(\mathcal{D}')
\end{tikzcd}$$

What is known/Spoilers

Complexity Class

- Code equivalence is not NP-hard
- Easy instance of PEP:
  Random codes!
- rand. reduction from PEP to GI
- reduction from LEP to PEP
  if $q \geq 5$: weakly self dual

Solvers

$$\begin{tikzcd}
\mathcal{C} \arrow[r] \arrow[dd, "\varphi"'] & \mathcal{D} & A_w(\mathcal{D}) \arrow[dd, leftrightarrow, "= ?"] & \\
 & & & \mathrm{ISD} \arrow[ul] \arrow[dl] \\
\mathcal{C}' \arrow[r] & \mathcal{D}' & A_w(\mathcal{D}') &
\end{tikzcd}$$

What is known/Spoilers

Complexity Class

- Code equivalence is not NP-hard
- Easy instance of PEP:
  Random codes!
- rand. reduction from PEP to GI
- reduction from LEP to PEP
  if $q \geq 5$: weakly self dual

Solvers

- all solvers have exponential cost
- use Information Set Decoding (ISD)

$$
\begin{tikzcd}
\mathcal{C} \arrow[r] \arrow[dd, "\varphi"'] & \mathcal{D} & A_w(\mathcal{D}) \arrow[dd, leftrightarrow, "= ?"] & \\
 & & & \mathrm{ISD} \arrow[ul] \arrow[dl] \\
\mathcal{C}' \arrow[r] & \mathcal{D}' & A_w(\mathcal{D}') &
\end{tikzcd}
$$

Methods/Buzzwords

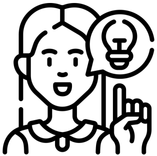
Tools we use/ what to expect in the project

- behavior of different hulls
- automorphism groups
- weight enumerators
- supports of subcodes

- code-based crypto
- algorithmics
- complexity theory
- ISD

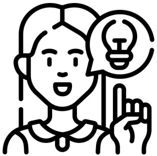
Questions

- Other reductions (not randomized)?
- Other easy instances?
- Other solvers: other invariants/subcodes?
- Other metrics?



Questions

- Other reductions (not randomized)?
- Other easy instances?
- Other solvers: other invariants/subcodes?
- Other metrics?



Notes on Code Equivalence

Thank you!

References

- M. Bardet, A. Otmani, and M. Saeed-Taha. "Permutation code equivalence is not harder than graph isomorphism when hulls are trivial", ISIT, 2019.
- N. Sendrier, D. E. Simos. "How easy is code equivalence over Fq?", WCC, 2013.
- E. Petrank, R.M. Roth. "Is code equivalence easy to decide?", IEEE TIT, 1997.
- N. Sendrier. "On the dimension of the hull.", SIAM Discr. Math., 1997.
- N. Sendrier. "Finding the permutation between equivalent linear codes: The support splitting algorithm.", IEEE TIT, 2000.
- I. Bouyukliev. "About the code equivalence.", Adv. in Coding and Crypto., 2007.
- L. Babai, P. Codenotti, J. Grochow, Y. Qiao. "Code equivalence and group isomorphism.", SIAM Discr. Math., 2011.
- A. Barenghi, J.F. Biasse, E. Persichetti, P. Santini. "On the computational hardness of the code equivalence problem in cryptography.", AMC, 2023.
- W. Beullens "Not enough LESS: an improved algorithm for solving code equivalence problems over F q.", Crypto, 2020.
- N. Sendrier, D. E. Simos. "The hardness of code equivalence over and its application to code-based cryptography.", PQCrypto, 2013.
- J. Leon. "Computing automorphism groups of error-correcting codes.", IEEE TIT, 1982.
- A. Barenghi, J.F. Biasse, E. Persichetti, P. Santini. "LESS-FM: fine-tuning signatures from the code equivalence problem.", PQCrypto, 2021.