Faster isomorphism testing of p-groups of Frattini class-2

Speaker: Chuanqi Zhang

Joint with Gábor Ivanyos, Euan Mendoza, Youming Qiao, and Xiaorui Sun

Centre for Quantum Software and Information
University of Technology Sydney, Australia

Theory of Computing Seminar at the University of Wisconsin-Madison

November 1, 2024

Faster isomorphism testing of p-groups of Frattini class-2

Speaker: Chuanqi Zhang

Joint with Gábor Ivanyos, Euan Mendoza, Youming Qiao, and Xiaorui Sun

Centre for Quantum Software and Information
University of Technology Sydney, Australia

Theory of Computing Seminar at the University of Wisconsin-Madison

November 1, 2024

This work has been accepted to FOCS 2024 and invited to the SICOMP special issue.

Outline

- Background of finite group isomorphism.
- From p-group isomorphism to 3-tensor isomorphism.
- Our main results and an overview of our algorithm.
- Summary and further directions.

Outline

- Background of finite group isomorphism.
- From p-group isomorphism to 3-tensor isomorphism.
- Our main results and an overview of our algorithm.
- Summary and further directions.

Outline

- Background of finite group isomorphism.
- From p-group isomorphism to 3-tensor isomorphism.
- Our main results and an overview of our algorithm.
- Summary and further directions.

Outline

- Background of finite group isomorphism.
- From p-group isomorphism to 3-tensor isomorphism.
- Our main results and an overview of our algorithm.
- Summary and further directions.

Group Isomorphism Problem

Problem (GROUP ISO)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

- First algorithm: $N^{\log N + O(1)}$ time attributed to Tarjan [Miller'78]

Group Isomorphism Problem

Problem (Group Iso)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$\mathbb{Z}_2$

$+$	0	1
0	0	1
1	1	0

$\cong$

S_2

$\circ$	e	s
e	e	s
s	s	e

- First algorithm: $N^{\log N + O(1)}$ time attributed to Tarjan [Miller'78]

Group Isomorphism Problem

Problem (GROUP ISO)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$\mathbb{Z}_2$

$+$	0	1
0	0	1
1	1	0

$\cong$

S_2

$\circ$	e	s
e	e	s
s	s	e

- First algorithm: $N^{\log N+O(1)}$ time attributed to Tarjan [Miller'78]

Group Isomorphism Problem

Problem (GROUP ISO)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$\mathbb{Z}_2$

$+$	0	1
0	0	1
1	1	0

$\cong$

S_2

$\circ$	e	s
e	e	s
s	s	e

- First algorithm: $N^{\log N + O(1)}$ time attributed to Tarjan [Miller'78]

Group Isomorphism Problem

Problem (GROUP ISO)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$$\begin{array}{c|cc} \multicolumn{3}{c}{\mathbb{Z}_2} \\ + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array} \quad \cong \quad \begin{array}{c|cc} \multicolumn{3}{c}{S_2} \\ \circ & e & s \\ \hline e & e & s \\ s & s & e \end{array}$$

- First algorithm: $N^{\log N + O(1)}$ time attributed to Tarjan [Miller'78]

Group Isomorphism Problem

Problem (Group Iso)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$\mathbb{Z}_2$

$+$	0	1
0	0	1
1	1	0

$\cong$

S_2

$\circ$	e	s
e	e	s
s	s	e

- First algorithm: $N^{\log N + O(1)}$ time attributed to Tarjan [Miller'78]
- Best known algorithm: $N^{\frac{1}{4}\log N + O(1)}$ time [Rosenbaum'13]
- Open question: $N^{\log N} \xrightarrow{?} N^{o(\log N)}$

Group Isomorphism Problem

Problem (GROUP ISO)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$$\begin{array}{c} \mathbb{Z}_2 \\ \begin{array}{c|cc} + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array} \end{array} \cong \begin{array}{c} S_2 \\ \begin{array}{c|cc} \circ & e & s \\ \hline e & e & s \\ s & s & e \end{array} \end{array}$$

- First algorithm: $N^{\log N + O(1)}$ time attributed to Tarjan [Miller'78]
- Best known algorithm: $N^{\frac{1}{4}\log N + O(1)}$ time [Rosenbaum'13]
- Open question: $N^{\log N} \xrightarrow{?} N^{o(\log N)}$

Group Isomorphism Problem

Problem (GROUP ISO)

Given the multiplication tables of two groups of finite order N, determine whether they are isomorphic.

$$
\begin{array}{c}
\mathbb{Z}_2 \\
\begin{array}{c|cc}
+ & 0 & 1 \\
\hline
0 & 0 & 1 \\
1 & 1 & 0
\end{array}
\end{array}
\quad \cong \quad
\begin{array}{c}
S_2 \\
\begin{array}{c|cc}
\circ & e & s \\
\hline
e & e & s \\
s & s & e
\end{array}
\end{array}
$$

- First algorithm: $N^{\log N+O(1)}$ time attributed to Tarjan [Miller'78]
- Best known algorithm: $N^{\frac{1}{4}\log N+O(1)}$ time [Rosenbaum'13]
- Open question: $N^{\log N} \xrightarrow{?} N^{o(\log N)}$

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	Group Iso	Graph Iso
Natural bound	$N^{\log N+O(1)}$ [Miller'78]	
Best known bound	$N^{\frac{1}{4}\log N+O(1)}$ [Rosenbaum'13]	

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	Group Iso	Graph Iso
Natural bound	$N^{\log N+O(1)}$ [Miller'78]	
Best known bound	$N^{\frac{1}{4}\log N+O(1)}$ [Rosenbaum'13]	

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	GROUP ISO	GRAPH ISO
Natural bound	$N^{\log N+O(1)}$ [Miller'78]	
Best known bound	$N^{\frac{1}{4}\log N+O(1)}$ [Rosenbaum'13]	

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	GROUP ISO	GRAPH ISO
Natural bound	$N^{\log N + O(1)}$ [Miller'78]	$N!$
Best known bound	$N^{\frac{1}{4}\log N + O(1)}$ [Rosenbaum'13]	$N^{O((\log N)^c)}$ [Babai'17]

GROUP ISO blocks us from an $N^{o(\log N)}$ time algorithm for GRAPH ISO [Babai'17]!

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	GROUP ISO	GRAPH ISO
Natural bound	$N^{\log N + O(1)}$ [Miller'78]	$N!$
Best known bound	$N^{\frac{1}{4}\log N + O(1)}$ [Rosenbaum'13]	$N^{O((\log N)^c)}$ [Babai'17]

GROUP ISO blocks us from an $N^{o(\log N)}$ time algorithm for GRAPH ISO [Babai'17]!

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	GROUP ISO	GRAPH ISO
Natural bound	$N^{\log N + O(1)}$ [Miller'78]	$N!$
Best known bound	$N^{\frac{1}{4}\log N + O(1)}$ [Rosenbaum'13]	$N^{O((\log N)^c)}$ [Babai'17]

GROUP ISO blocks us from an $N^{o(\log N)}$ time algorithm for GRAPH ISO [Babai'17]!

Note that $c > 1$ in Babai's algorithm.

Group Isomorphism vs. Graph Isomorphism

Similarity

- Both are extensively studied since 1970s.
- Both are neither known to be in P nor to be NP-complete.
- Best algorithms for both problems have a quasi-polynomial running time.

Differences

	GROUP ISO	GRAPH ISO
Natural bound	$N^{\log N+O(1)}$ [Miller'78]	$N!$
Best known bound	$N^{\frac{1}{4}\log N+O(1)}$ [Rosenbaum'13]	$N^{O((\log N)^c)}$ [Babai'17]

GROUP ISO blocks us from an $N^{o(\log N)}$ time algorithm for GRAPH ISO [Babai'17]!

Note that $c > 1$ in Babai's algorithm.

Through the lens of GROUP ISO testing

- Theoretical computer science: complexity in the worst case

- Computational group theory: practical algorithms (as in Magma or GAP)

- Cryptography: protocols based on isomorphism problems

All of these areas can give us good motivation to study GROUP ISO testing ;)

Through the lens of Group Iso testing

- Theoretical computer science: complexity in the worst case
- Computational group theory: practical algorithms (as in Magma or GAP)
- Cryptography: protocols based on isomorphism problems

All of these areas can give us good motivation to study Group Iso testing ;)

Through the lens of Group Iso testing

- Theoretical computer science: complexity in the worst case
- Computational group theory: practical algorithms (as in Magma or GAP)
- Cryptography: protocols based on isomorphism problems
  - Several schemes have been submitted to the NIST call for post-quantum digital signatures, such as *ALTEQ*, *MEDS*, and *LESS*.

All of these areas can give us good motivation to study Group Iso testing :)

Through the lens of Group Iso testing

- Theoretical computer science: complexity in the worst case
- Computational group theory: practical algorithms (as in Magma or GAP)
- Cryptography: protocols based on isomorphism problems
  - Several schemes have been submitted to the NIST call for post-quantum digital signatures, such as *ALTEQ*, *MEDS*, and *LESS*.

All of these areas can give us good motivation to study Group Iso testing ;)

Through the lens of GROUP ISO testing

- Theoretical computer science: complexity in the worst case
- Computational group theory: practical algorithms (as in Magma or GAP)
- Cryptography: protocols based on isomorphism problems
  - Several schemes have been submitted to the NIST call for post-quantum digital signatures, such as *ALTEQ*, *MEDS*, and *LESS*.

All of these areas can give us good motivation to study GROUP ISO testing :)

Recent breakthrough in Group Iso of a special class

Theorem (Sun'23)

Given two p-groups of class-2 and exponent p of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{5/6})}$ *to decide whether they are isomorphic.*

Why this class of groups?

1. The isomorphism testing between p-groups of class-2 and exponent p is a major bottleneck for Group Iso.

Definition (p-groups of class-2 and exponent p)

For prime p, we say a p-group G is of *class-2 and exponent p*, if

- every $g \in G$ satisfies that $g^p = \mathrm{id}$, and
- $[G, [G, G]]$ only contains the identity element.

2. The isomorphism testing between p-groups of class-2 and exponent p can reduce to the isomorphism testing between two 3-tensors.

Recent breakthrough in Group Iso of a special class

Theorem (Sun'23)

Given two p-groups of class-2 and exponent p of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{5/6})}$ *to decide whether they are isomorphic.*

Why this class of groups?

1. The isomorphism testing between p-groups of class-2 and exponent p is a major bottleneck for Group Iso.

Definition (p-groups of class-2 and exponent p)

For prime p, we say a p-group G is of *class-2 and exponent p*, if

- every $g \in G$ satisfies that $g^p = \text{id}$, and
- $[G, [G, G]]$ only contains the identity element.

2. The isomorphism testing between p-groups of class-2 and exponent p can reduce to the isomorphism testing between two 3-tensors.

Recent breakthrough in GROUP ISO of a special class

Theorem (Sun'23)

Given two p-groups of class-2 and exponent p of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{5/6})}$ to decide whether they are isomorphic.

Why this class of groups?

1. The isomorphism testing between p-groups of class-2 and exponent p is a major bottleneck for GROUP ISO.

Definition (p-groups of class-2 and exponent p)

For prime p, we say a p-group G is of *class-2 and exponent p*, if

- every $g \in G$ satisfies that $g^p = \text{id}$, and
- $[G, [G, G]]$ only contains the identity element.

2. The isomorphism testing between p-groups of class-2 and exponent p can reduce to the isomorphism testing between two 3-tensors.

Recent breakthrough in Group Iso of a special class

Theorem (Sun'23)

Given two p-groups of class-2 and exponent p of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{5/6})}$ to decide whether they are isomorphic.

Why this class of groups?

1. The isomorphism testing between p-groups of class-2 and exponent p is a major bottleneck for Group Iso.

Definition (p-groups of class-2 and exponent p)

For prime p, we say a p-group G is of *class-2 and exponent $p*$, if

- every $g \in G$ satisfies that $g^p = \mathrm{id}$, and
- $[G, [G, G]]$ only contains the identity element.

2. The isomorphism testing between p-groups of class-2 and exponent p can reduce to the isomorphism testing between two 3-tensors.

Recent breakthrough in GROUP ISO of a special class

Theorem (Sun'23)

Given two p-groups of class-2 and exponent p of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{5/6})}$ *to decide whether they are isomorphic.*

Why this class of groups?

1. The isomorphism testing between p-groups of class-2 and exponent p is a major bottleneck for GROUP ISO.

Definition (p-groups of class-2 and exponent p)

For prime p, we say a p-group G is of *class-2 and exponent* p, if

- every $g \in G$ satisfies that $g^p = \mathrm{id}$, and
- $[G, [G, G]]$ only contains the identity element.

2. The isomorphism testing between p-groups of class-2 and exponent p can reduce to the isomorphism testing between two 3-tensors.

Recent breakthrough in GROUP ISO of a special class

Theorem (Sun'23)

Given two p-groups of class-2 and exponent p of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{5/6})}$ *to decide whether they are isomorphic.*

Why this class of groups?

1. The isomorphism testing between p-groups of class-2 and exponent p is a major bottleneck for GROUP ISO.

Definition (p-groups of class-2 and exponent p)

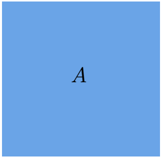
For prime p, we say a p-group G is of *class-2 and exponent* p, if

- every $g \in G$ satisfies that $g^p = \mathrm{id}$, and
- $[G, [G, G]]$ only contains the identity element.

2. The isomorphism testing between p-groups of class-2 and exponent p can reduce to the isomorphism testing between two 3-tensors.

Tensor Isomorphism Problem

- Tensors are multi-way arrays, e.g., 2-tensor $A = (a_{i,j})_n$ is an $n \times n$ matrix:

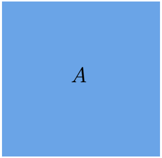


- Similarly, 3-tensors are arrays with 3 indices, like a cube with matrix slices:

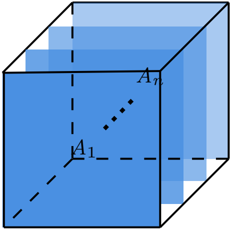
- More generally, we can define d-tensors, but d-Tensor Iso is as hard as 3-Tensor Iso. [Grochow-Qiao'23]

Tensor Isomorphism Problem

- Tensors are multi-way arrays, e.g., 2-tensor $A = (a_{i,j})_n$ is an $n \times n$ matrix:



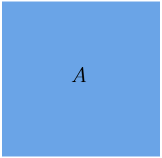
- Similarly, 3-tensors are arrays with 3 indices, like a cube with matrix slices:



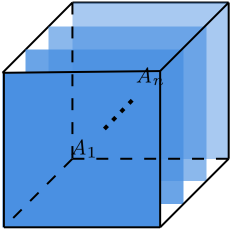
- More generally, we can define d-tensors, but d-Tensor Iso is as hard as 3-Tensor Iso. [Grochow-Qiao'23]

Tensor Isomorphism Problem

- Tensors are multi-way arrays, e.g., 2-tensor $A = (a_{i,j})_n$ is an $n \times n$ matrix:



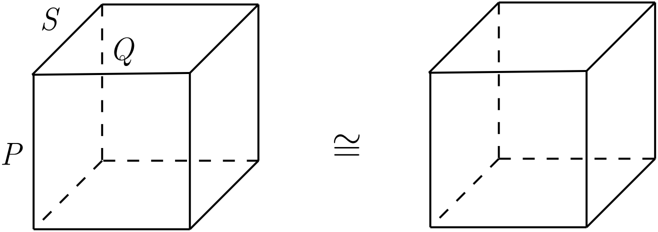
- Similarly, 3-tensors are arrays with 3 indices, like a cube with matrix slices:



- More generally, we can define d-tensors, but d-Tensor Iso is as hard as 3-Tensor Iso. [Grochow-Qiao'23]

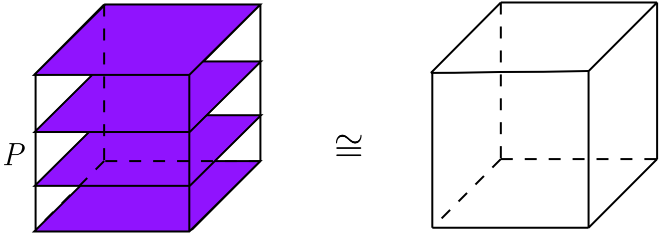
Tensor Isomorphism Problem

Isomorphism for 3-tensors under three invertible matrices P, Q, and S:



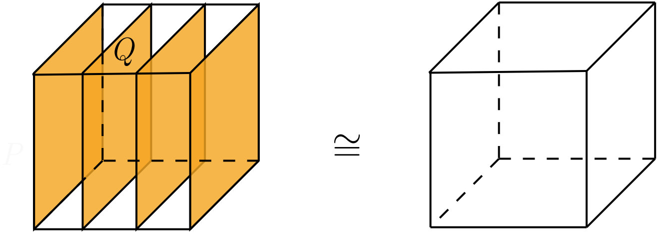
Tensor Isomorphism Problem

Isomorphism for 3-tensors under three invertible matrices P, Q, and S:



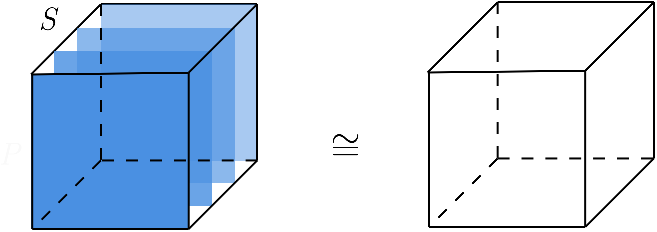
Tensor Isomorphism Problem

Isomorphism for 3-tensors under three invertible matrices P, Q, and S:



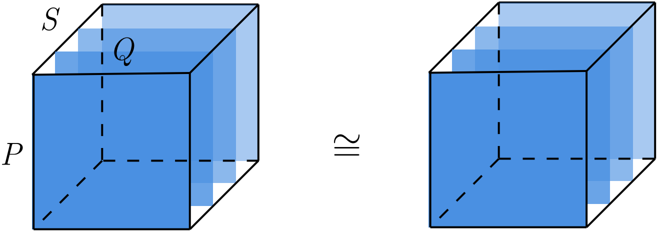
Tensor Isomorphism Problem

Isomorphism for 3-tensors under three invertible matrices P, Q, and S:



Tensor Isomorphism Problem

Isomorphism for 3-tensors under three invertible matrices P, Q, and S:

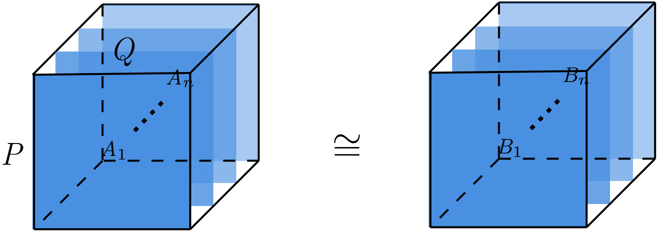


$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Tensor Isomorphism Problem

Isomorphism for 3-tensors under three invertible matrices P, Q, and S:



$$\mathrm{span}\{P\underset{\searrow}{A_1}Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Tensor Isomorphism Problem

Definition (Linear span of matrices)

Let $\{B_i : i \in [n]\}$ be a set of matrices over $\mathbb{F}_q$. Then

$$\mathrm{span}\{B_i : i \in [n]\} := \{\textstyle\sum_{i=1}^{n} c_i B_i : c_i \in \mathbb{F}_q\}.$$

Problem (Equivalence testing of 3-tensors)

Given two $n \times n \times n$ tensors over $\mathbb{F}_q$ whose frontal slices are $\{A_i : i \in [n]\}$ and $\{B_i : i \in [n]\}$, respectively. Determine if they are equivalent, i.e., if there exist two invertible matrices P and Q such that

$$\mathrm{span}\{B_i : i \in [n]\} = \mathrm{span}\{PA_iQ : i \in [n]\}.$$

Problem (Congruence testing of 3-tensors)

Given two $n \times n \times n$ tensors over $\mathbb{F}_q$ whose frontal slices are $\{A_i : i \in [n]\}$ and $\{B_i : i \in [n]\}$, respectively. Determine if they are congruent, i.e., if there exist one invertible matrix T such that

$$\mathrm{span}\{B_i : i \in [n]\} = \mathrm{span}\{T^t A_i T : i \in [n]\}.$$

Tensor Isomorphism Problem

Definition (Linear span of matrices)

Let $\{B_i : i \in [n]\}$ be a set of matrices over $\mathbb{F}_q$. Then

$$\text{span}\{B_i : i \in [n]\} := \{\textstyle\sum_{i=1}^{n} c_i B_i : c_i \in \mathbb{F}_q\}.$$

Problem (Equivalence testing of 3-tensors)

Given two $n \times n \times n$ tensors over $\mathbb{F}_q$ whose frontal slices are $\{A_i : i \in [n]\}$ and $\{B_i : i \in [n]\}$, respectively. Determine if they are equivalent, i.e., if there exist two invertible matrices P and Q such that

$$\text{span}\{B_i : i \in [n]\} = \text{span}\{PA_iQ : i \in [n]\}.$$

Problem (Congruence testing of 3-tensors)

Given two $n \times n \times n$ tensors over $\mathbb{F}_q$ whose frontal slices are $\{A_i : i \in [n]\}$ and $\{B_i : i \in [n]\}$, respectively. Determine if they are congruent, i.e., if there exist one invertible matrix T such that

$$\text{span}\{B_i : i \in [n]\} = \text{span}\{T^t A_i T : i \in [n]\}.$$

Tensor Isomorphism Problem

Definition (Linear span of matrices)

Let $\{B_i : i \in [n]\}$ be a set of matrices over $\mathbb{F}_q$. Then

$$\mathrm{span}\{B_i : i \in [n]\} := \{\textstyle\sum_{i=1}^{n} c_i B_i : c_i \in \mathbb{F}_q\}.$$

Problem (Equivalence testing of 3-tensors)

Given two $n \times n \times n$ *tensors over* $\mathbb{F}_q$ *whose frontal slices are* $\{A_i : i \in [n]\}$ *and* $\{B_i : i \in [n]\}$*, respectively. Determine if they are equivalent, i.e., if there exist two invertible matrices* P *and* Q *such that*

$$\mathrm{span}\{B_i : i \in [n]\} = \mathrm{span}\{PA_iQ : i \in [n]\}.$$

Problem (Congruence testing of 3-tensors)

Given two $n \times n \times n$ *tensors over* $\mathbb{F}_q$ *whose frontal slices are* $\{A_i : i \in [n]\}$ *and* $\{B_i : i \in [n]\}$*, respectively. Determine if they are congruent, i.e., if there exist one invertible matrix* T *such that*

$$\mathrm{span}\{B_i : i \in [n]\} = \mathrm{span}\{T^{t} A_i T : i \in [n]\}.$$

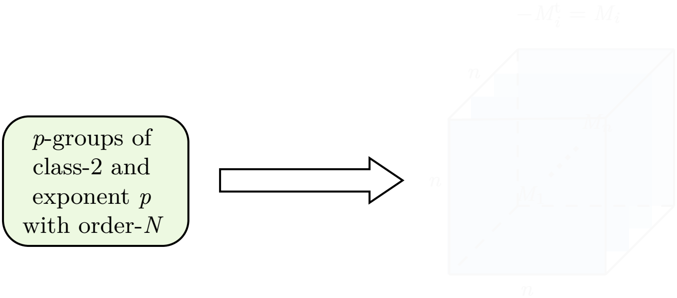
From p-GROUP ISO to 3-TENSOR ISO


p-groups of class-2 and exponent p with order-N

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3-tensors over $\mathbb{F}_p$ are congruent.

From p-GROUP ISO to 3-TENSOR ISO

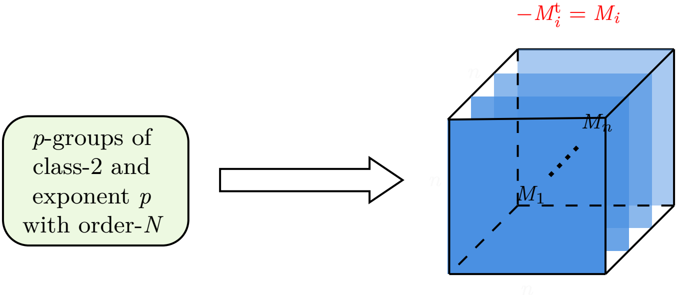


p-groups of class-2 and exponent p with order-N

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3-tensors over $\mathbb{F}_p$ are congruent.

From p-Group Iso to 3-Tensor Iso

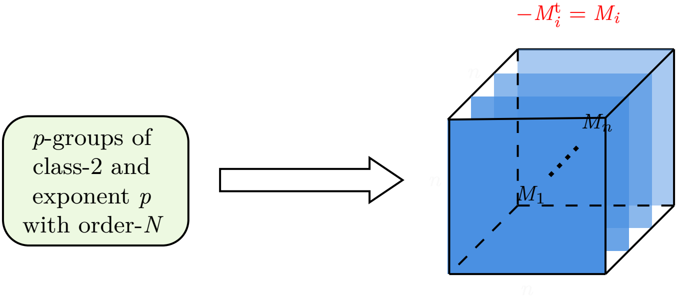


p-groups of class-2 and exponent p with order-N

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3-tensors over $\mathbb{F}_p$ are congruent.

From p-GROUP ISO to 3-TENSOR ISO



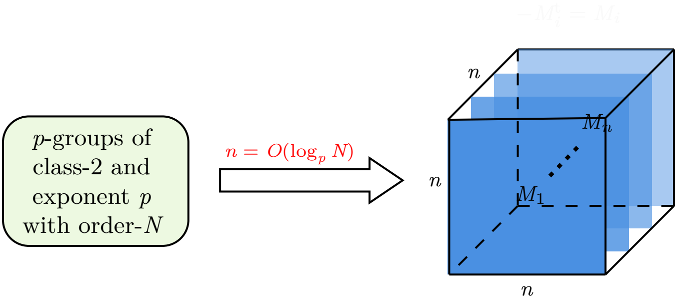
p-groups of
class-2 and
exponent p
with order-N

$-M_i^{\mathrm{t}} = M_i$

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3*-tensors over $\mathbb{F}_p$ are congruent.*

From p-GROUP ISO to 3-TENSOR ISO



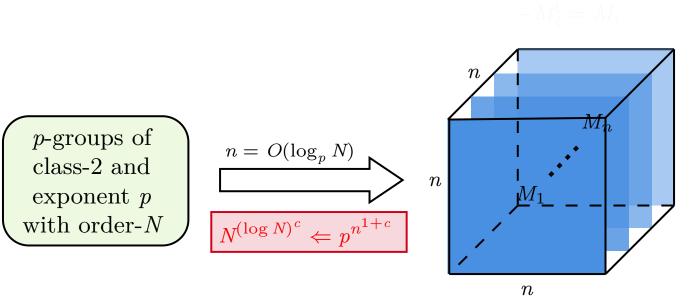
p-groups of class-2 and exponent p with order-N

$n = O(\log_p N)$

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3*-tensors over* $\mathbb{F}_p$ *are congruent.*

From p-GROUP ISO to 3-TENSOR ISO



p-groups of class-2 and exponent p with order-N

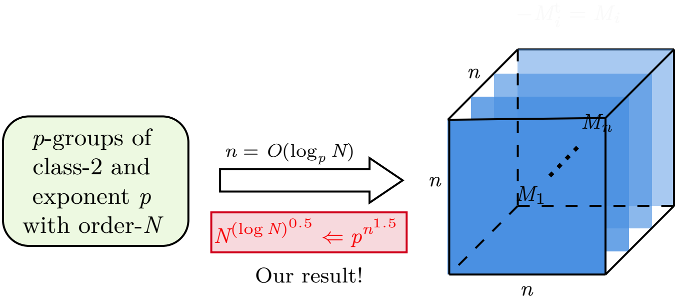
$n = O(\log_p N)$

$N^{(\log N)^c} \Leftarrow p^{n^{1+c}}$

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3-tensors over $\mathbb{F}_p$ are congruent.

From p-GROUP ISO to 3-TENSOR ISO



p-groups of class-2 and exponent p with order-N

$n = O(\log_p N)$

$N^{(\log N)^{0.5}} \Leftarrow p^{n^{1.5}}$

Our result!

Theorem (Baer's correspondence)

Two p-groups of class-2 and exponent p are isomorphic if and only if their associated skew-symmetric 3-tensors over $\mathbb{F}_p$ are congruent.

Previous work and our main result

- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8}\cdot\log q)}$ [Sun'23]
- Our improvement:

$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Previous work and our main result

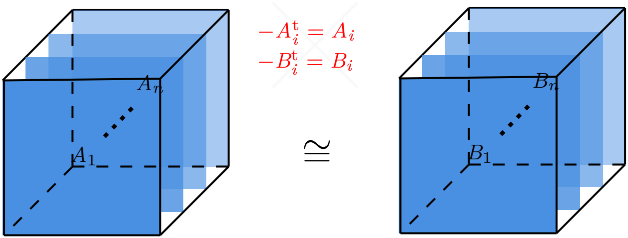
- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8} \cdot \log q)}$ [Sun'23]
- Our improvement:

$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Previous work and our main result

- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8} \cdot \log q)}$ [Sun'23]
  - addressed the congruence testing of skew-symmetric 3-tensors.
  -
- Our improvement:

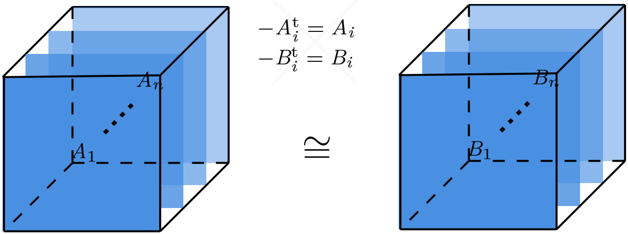


$$\mathrm{span}\{T^t A_1 T, \cdots, T^t A_n T\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Previous work and our main result

- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8}\cdot\log q)}$ [Sun'23]
    - addressed the congruence testing of skew-symmetric 3-tensors.
    - improved the isomorphism testing of *p*-groups of class-2 and exponent *p*.
- Our improvement:

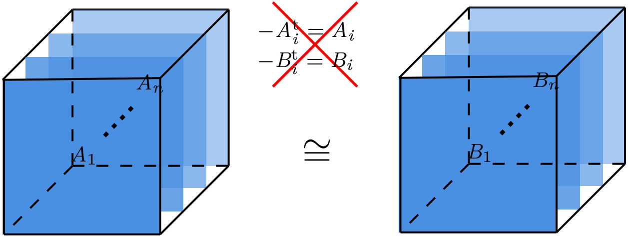


$$\mathrm{span}\{T^{\mathrm{t}} A_1 T, \cdots, T^{\mathrm{t}} A_n T\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Previous work and our main result

- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8} \cdot \log q)}$ [Sun'23]
  - addressed a special 3-Tensor Iso problem reducible to our problem.
  -
- Our improvement: $q^{\tilde{O}(n^{1.5})}$ for the equivalence testing of general 3-tensors



$\cancel{-A_i^{\mathrm{t}} = A_i}$

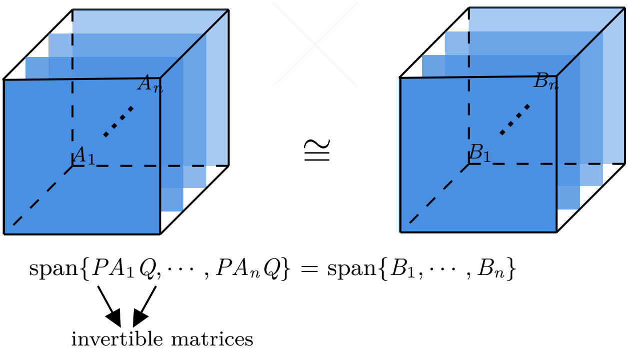
$\cancel{-B_i^{\mathrm{t}} = B_i}$

$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

Previous work and our main result

- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8} \cdot \log q)}$ [Sun'23]
  - addressed a special 3-Tensor Iso problem reducible to our problem.
  - improved the isomorphism testing of a subclass of our underlying groups.*
- Our improvement: $q^{\tilde{O}(n^{1.5})}$ for the equivalence testing of general 3-tensors



$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

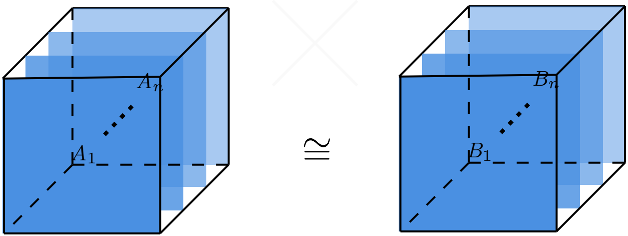
invertible matrices

---

*We extend to *p*-groups of Frattini class-2 by the results in [Higman'60, Grochow-Qiao'24].

Previous work and our main result

- Natural upper bound: $q^{O(n^2)}$ (known since at least 1970's)
- Sun's breakthrough: $q^{O(n^{1.8} \cdot \log q)}$ [Sun'23]
  - addressed a special 3-Tensor Iso problem reducible to our problem.
  - improved the isomorphism testing of a subclass of our underlying groups.*
- Our improvement: $q^{\tilde{O}(n^{1.5})}$ for the equivalence testing of general 3-tensors



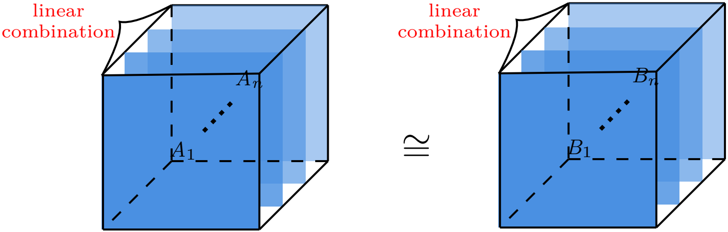
$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

invertible matrices

---

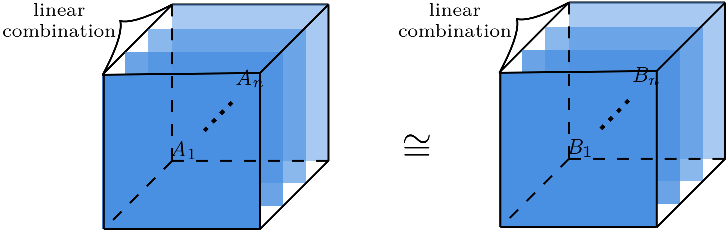
*We extend to p-groups of Frattini class-2 by the results in [Higman'60, Grochow-Qiao'24].

Overall strategy: from TENSOR ISO to TUPLE ISO



$$\text{span}\{PA_1Q, \cdots, PA_nQ\} = \text{span}\{B_1, \cdots, B_n\}$$

Overall strategy: from TENSOR ISO to TUPLE ISO

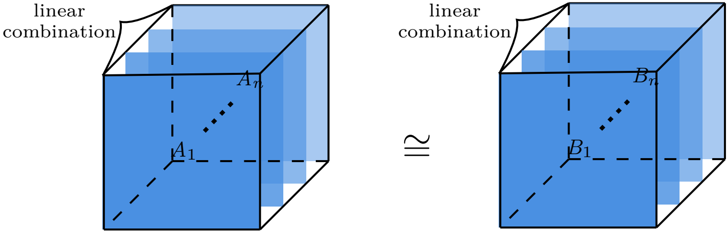


$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

Overall strategy: reduce the equivalence testing of 3-tensors to the congruence testing of matrix tuples, which is solvable in polynomial time [Ivanyos-Qiao'19].

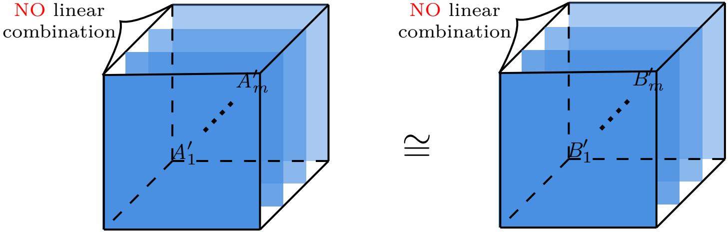
$$(T^t A'_1 T, \cdots, T^t A'_m T) = (B'_1, \cdots, B'_m)$$

Overall strategy: from TENSOR ISO to TUPLE ISO



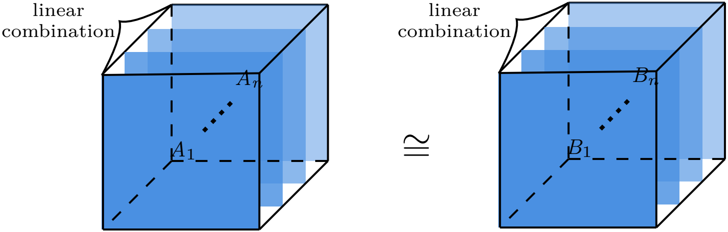
$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

Overall strategy: reduce the equivalence testing of 3-tensors to the congruence testing of matrix tuples, which is solvable in polynomial time [Ivanyos-Qiao'19].



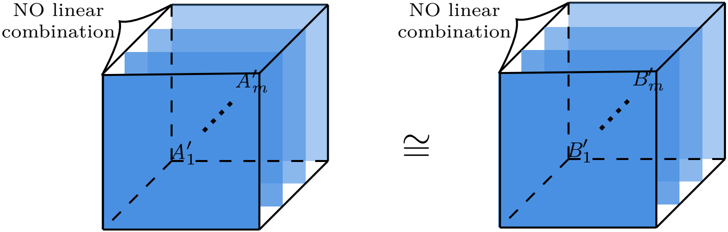
$$(T^t A'_1 T, \cdots, T^t A'_m T) = (B'_1, \cdots, B'_m)$$

Overall strategy: from TENSOR ISO to TUPLE ISO



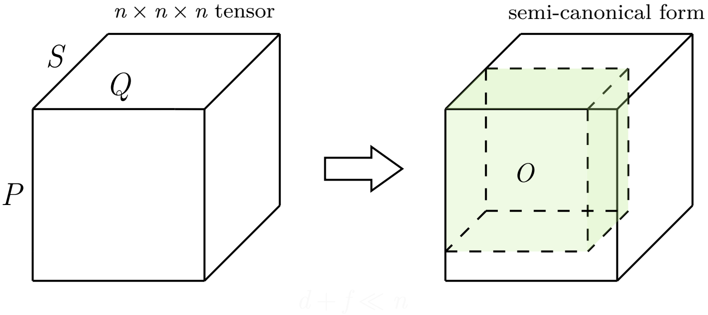
$$\mathrm{span}\{PA_1Q, \cdots, PA_nQ\} = \mathrm{span}\{B_1, \cdots, B_n\}$$

Overall strategy: reduce the equivalence testing of 3-tensors to the congruence testing of matrix tuples, which is solvable in polynomial time [Ivanyos-Qiao'19].

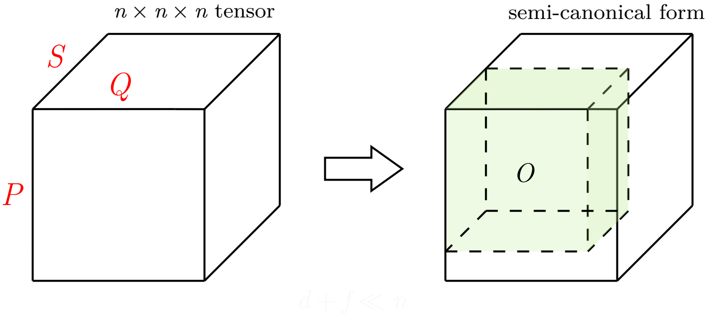


$$(T^t A'_1 T, \cdots, T^t A'_m T) = (B'_1, \cdots, B'_m)$$

Bridge: semi-canonical forms of equivalent tensors

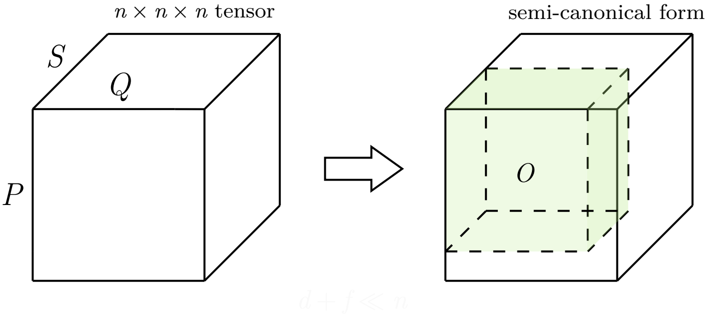


Bridge: semi-canonical forms of equivalent tensors



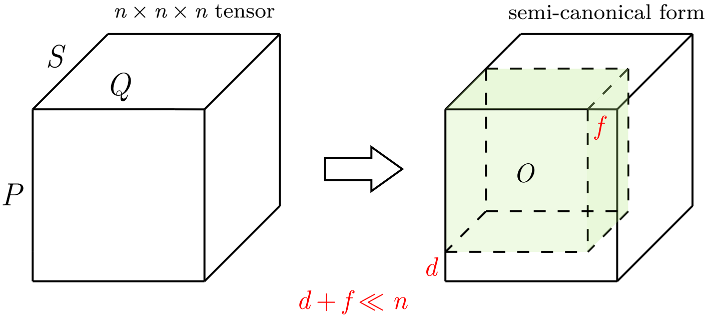
- We first make the 3-tensors in a semi-canonical form by applying P, Q and S, and then construct matrix tuples from the semi-canonical 3-tensors.
- The margins are supposed to be small, to reduce the cost of further enumeration of the possible action matrices.
- The margin for the third direction, while can be large, is ‘fixed’ somehow.

Bridge: semi-canonical forms of equivalent tensors



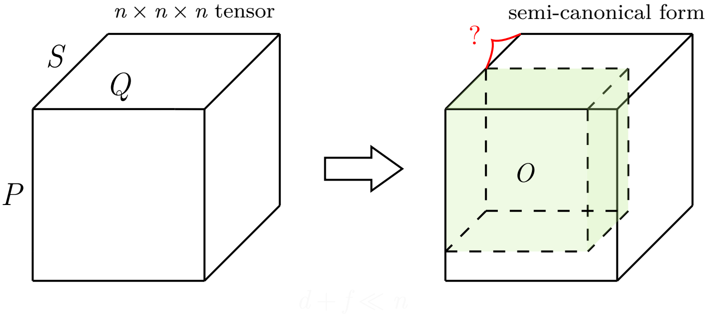
- We first make the 3-tensors in a semi-canonical form by applying P, Q and S, and then construct matrix tuples from the semi-canonical 3-tensors.
- The margins are supposed to be small, to reduce the cost of further enumeration of the possible action matrices.
- The margin for the third direction, while can be large, is 'fixed' somehow.

Bridge: semi-canonical forms of equivalent tensors



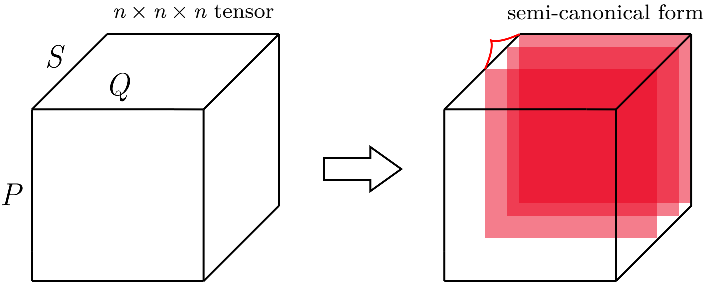
- We first make the 3-tensors in a semi-canonical form by applying P, Q and S, and then construct matrix tuples from the semi-canonical 3-tensors.
- The margins are supposed to be small, to reduce the cost of further enumeration of the possible action matrices.
- The margin for the third direction, while can be large, is 'fixed' somehow.

Bridge: semi-canonical forms of equivalent tensors



- We first make the 3-tensors in a semi-canonical form by applying P, Q and S, and then construct matrix tuples from the semi-canonical 3-tensors.
- The margins are supposed to be small, to reduce the cost of further enumeration of the possible action matrices.
- The margin for the third direction, while can be large, is ‘fixed’ somehow.

Bridge: semi-canonical forms of equivalent tensors

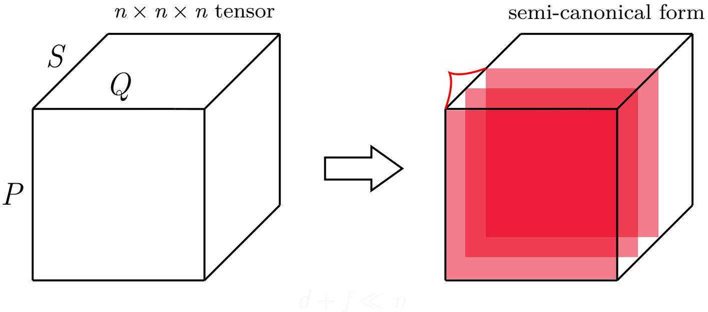


$d + f \ll n$

Two key techniques:

1. Refinement: fix rear slices and leave the frontal to span a low-rank space
2. Low-rank characterization: make a big zero block on the low-rank slices

Bridge: semi-canonical forms of equivalent tensors



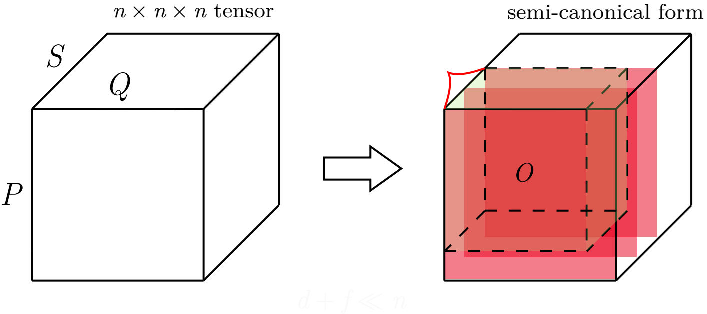
Two key techniques:

1. Refinement: fix rear slices and leave the frontal to span a low-rank space*
2. Low-rank characterization: make a big zero block on the low-rank slices

---

*A linear space of matrices is of low rank, if every matrix in it is of low rank.

Bridge: semi-canonical forms of equivalent tensors



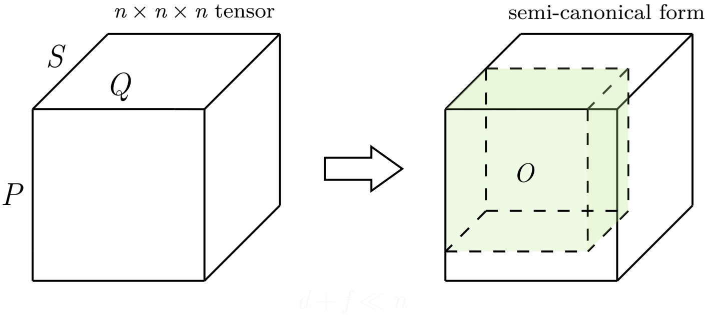
Two key techniques:

1. Refinement: fix rear slices and leave the frontal to span a low-rank space*
2. Low-rank characterization: make a big zero block on the low-rank slices

---

*A linear space of matrices is of low rank, if every matrix in it is of low rank.

Bridge: semi-canonical forms of equivalent tensors



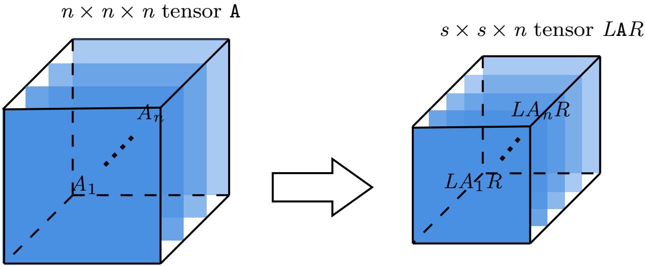
Two key techniques:

1. Refinement: fix rear slices and leave the frontal to span a low-rank space*
2. Low-rank characterization: make a big zero block on the low-rank slices

---

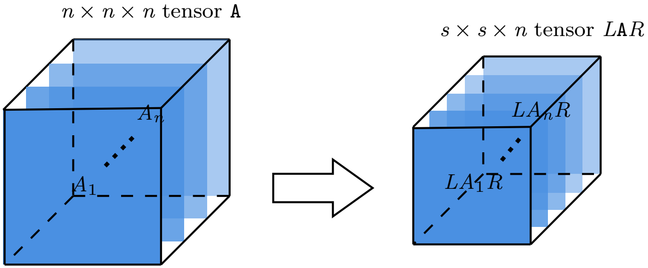
*A linear space of matrices is of low rank, if every matrix in it is of low rank.

A special case: canonicalization by compression



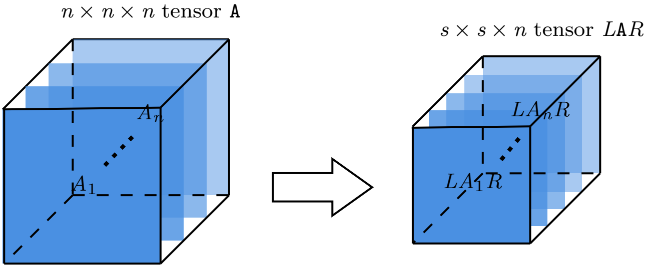
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors $\mathtt{A}$ and $\mathtt{B}$.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

A special case: canonicalization by compression



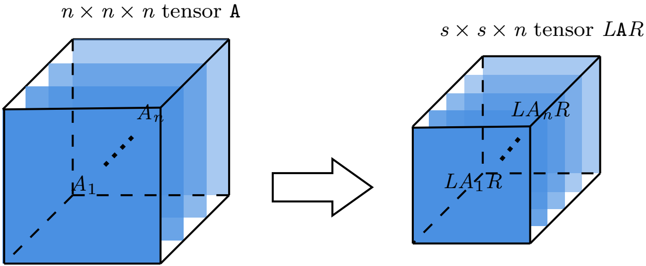
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors `A` and `B`.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

A special case: canonicalization by compression



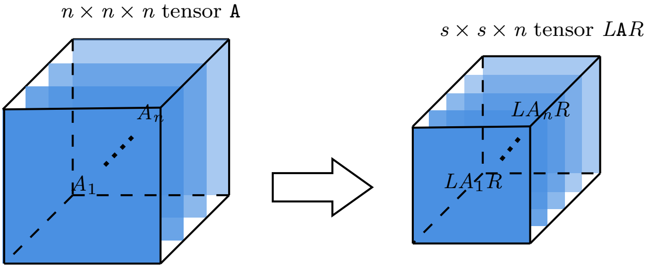
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors $\mathtt{A}$ and $\mathtt{B}$.
  - Compute the canonical basis of $L\mathtt{A}R$.
  - Enumerate such matrices L' and R' for $\mathtt{B}$, which costs $q^{O(ns)}$.
  - Compute the canonical basis of $L'\mathtt{B}R'$ and compare it to that of $L\mathtt{A}R$.
  - The correspondence between L, L' and R, R' gives the desired isomorphism.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

A special case: canonicalization by compression



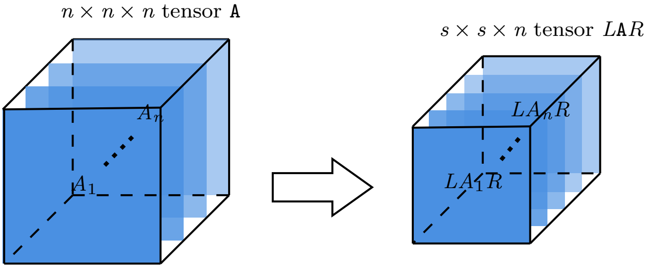
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors $\mathtt{A}$ and $\mathtt{B}$.
    - Compute the canonical basis of $L\mathtt{A}R$.
    - Enumerate such matrices L' and R' for $\mathtt{B}$, which costs $q^{O(ns)}$.
    - Compute the canonical basis of $L'\mathtt{B}R'$ and compare it to that of $L\mathtt{A}R$.
    - The correspondence between L, L' and R, R' gives the desired isomorphism.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

A special case: canonicalization by compression



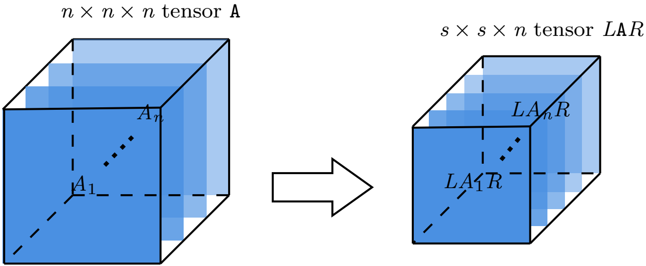
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors $\mathtt{A}$ and $\mathtt{B}$.
  - Compute the canonical basis of $L\mathtt{A}R$.
  - Enumerate such matrices L' and R' for $\mathtt{B}$, which costs $q^{O(ns)}$.
  - Compute the canonical basis of $L'\mathtt{B}R'$ and compare it to that of $L\mathtt{A}R$.
  - The correspondence between L, L' and R, R' gives the desired isomorphism.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

A special case: canonicalization by compression



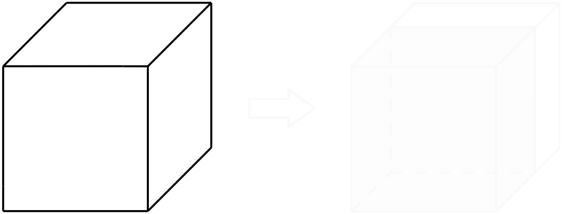
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors A and B.
    - Compute the canonical basis of $L\mathtt{A}R$.
    - Enumerate such matrices L' and R' for B, which costs $q^{O(ns)}$.
    - Compute the canonical basis of $L'\mathtt{B}R'$ and compare it to that of $L\mathtt{A}R$.
    - The correspondence between L, L' and R, R' gives the desired isomorphism.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

A special case: canonicalization by compression



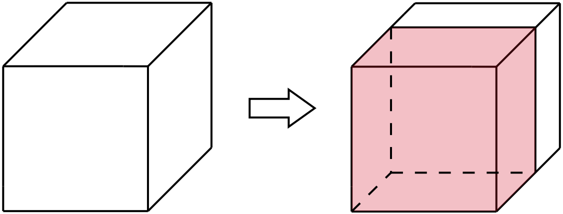
- Assume we can apply $L \in \mathrm{GL}(s \times n, \mathbb{F}_q)$ and $R \in \mathrm{GL}(n \times s, \mathbb{F}_q)$ such that $LA_1R, \cdots, LA_nR \in \mathrm{M}(s \times s, \mathbb{F}_q)$ are linearly independent.
- Then there is a quick algorithm to test the isomorphism between two 3-tensors $\mathtt{A}$ and $\mathtt{B}$.
  - Compute the canonical basis of $L\mathtt{A}R$.
  - Enumerate such matrices L' and R' for $\mathtt{B}$, which costs $q^{O(ns)}$.
  - Compute the canonical basis of $L'\mathtt{B}R'$ and compare it to that of $L\mathtt{A}R$.
  - The correspondence between L, L' and R, R' gives the desired isomorphism.
- What if $LAR = 0$ for some non-zero $A \in \mathrm{span}\{A_i : i \in [n]\}$?

Technique 1: refine the frontal slices



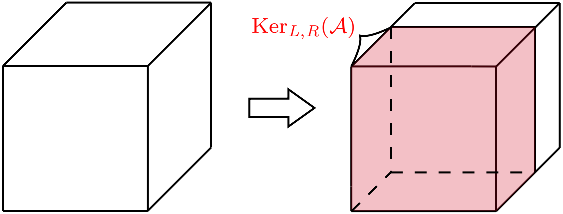
- Given a 3-tensor `A` whose frontal slices span $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$.
- Basic idea: sort the basis matrices (subject to choices of L, R) such that
- Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Technique 1: refine the frontal slices



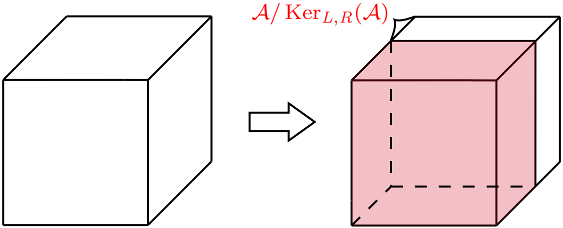
- Given a 3-tensor `A` whose frontal slices span $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$.
- Basic idea: sort the basis matrices (subject to choices of L, R) such that
  - the first ones span $\mathrm{Ker}_{L,R}(\mathcal{A}) := \mathrm{span}\{A \in \mathcal{A} \mid LAR = 0\}$, and
  - the remaining ones form a canonical basis of the quotient space $\mathcal{A}/\,\mathrm{Ker}_{L,R}(\mathcal{A})$.
- Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Technique 1: refine the frontal slices



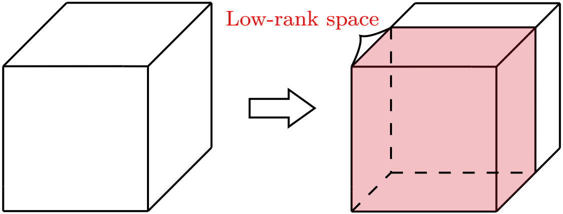
- Given a 3-tensor `A` whose frontal slices span $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$.
- Basic idea: sort the basis matrices (subject to choices of L, R) such that
    - the first ones span $\mathrm{Ker}_{L,R}(\mathcal{A}) := \mathrm{span}\{A \in \mathcal{A} \mid LAR = 0\}$, and
    - the remaining ones form a canonical basis of the quotient space $\mathcal{A}/\mathrm{Ker}_{L,R}(\mathcal{A})$.
- Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Technique 1: refine the frontal slices



- Given a 3-tensor `A` whose frontal slices span $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$.
- Basic idea: sort the basis matrices (subject to choices of L, R) such that
  - the first ones span $\operatorname{Ker}_{L,R}(\mathcal{A}) := \operatorname{span}\{A \in \mathcal{A} \mid LAR = 0\}$, and
  - the remaining ones form a canonical basis of the quotient space $\mathcal{A}/\operatorname{Ker}_{L,R}(\mathcal{A})$.
- Advantage: $\operatorname{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Technique 1: refine the frontal slices



- Given a 3-tensor `A` whose frontal slices span $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$.
- Basic idea: sort the basis matrices (subject to choices of L, R) such that
    - the first ones span $\operatorname{Ker}_{L,R}(\mathcal{A}) := \operatorname{span}\{A \in \mathcal{A} \mid LAR = 0\}$, and
    - the remaining ones form a canonical basis of the quotient space $\mathcal{A}/\operatorname{Ker}_{L,R}(\mathcal{A})$.
- Advantage: $\operatorname{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Technique 1: refine the frontal slices of a 3-tensor

Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Lemma (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Let $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$ *be a matrix subspace of dimension* n. *Then with at least probability of* $1 - \frac{1}{q^r}$, $\mathrm{Ker}_{L,R}(\mathcal{A})$ *consists of matrices of rank* $\leq r$ *for uniformly randomly sampled* $L \in \mathrm{M}(s \times n, \mathbb{F}_q)$ *and* $R \in \mathrm{M}(n \times s, \mathbb{F}_q)$.

Why is this an advantage?

Technique 1: refine the frontal slices of a 3-tensor

Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Lemma (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Let $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$ *be a matrix subspace of dimension* n. *Fix some* $r \in [n]$, *and let*

$$s = \lceil 3 \cdot \max\{\frac{n}{r}, r\} \rceil.$$

Then with at least probability of $1 - \frac{1}{q^r}$, $\mathrm{Ker}_{L,R}(\mathcal{A})$ *consists of matrices of rank* $\leq r$ *for uniformly randomly sampled* $L \in \mathrm{M}(s \times n, \mathbb{F}_q)$ *and* $R \in \mathrm{M}(n \times s, \mathbb{F}_q)$.

Why is this an advantage?

Technique 1: refine the frontal slices of a 3-tensor

Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Lemma (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Let $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$ *be a matrix subspace of dimension* n. *Let* $r = \sqrt{n}$ *and*

$$s = O(\sqrt{n}).$$

Then with at least probability of $1 - \frac{1}{q^r}$, $\mathrm{Ker}_{L,R}(\mathcal{A})$ *consists of matrices of rank* $\leq r$ *for uniformly randomly sampled* $L \in \mathrm{M}(s \times n, \mathbb{F}_q)$ *and* $R \in \mathrm{M}(n \times s, \mathbb{F}_q)$.

Why is this an advantage?

Technique 1: refine the frontal slices of a 3-tensor

Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Lemma (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Let $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$ *be a matrix subspace of dimension* n. *Let* $r = \sqrt{n}$ *and*

$$s = O(\sqrt{n}).$$

Then with at least probability of $1 - \frac{1}{q^r}$, $\mathrm{Ker}_{L,R}(\mathcal{A})$ *consists of matrices of rank* $\leq r$ *for uniformly randomly sampled* $L \in \mathrm{M}(s \times n, \mathbb{F}_q)$ *and* $R \in \mathrm{M}(n \times s, \mathbb{F}_q)$.

Again, to find L', R' such that `B` is refined correspondingly to `A`, we still need to enumerate all L', R' in the same size, which costs $q^{O(ns)}$. Why is this an advantage?

Technique 1: refine the frontal slices of a 3-tensor

Advantage: $\mathrm{Ker}_{L,R}(\mathcal{A})$ is a low-rank subspace with a high probability.

Lemma (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

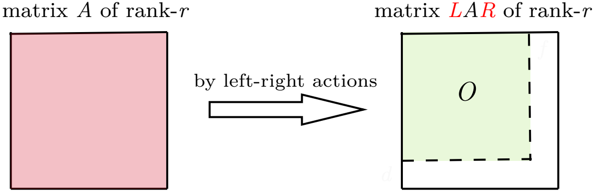
Let $\mathcal{A} \leq \mathrm{M}(n, \mathbb{F}_q)$ *be a matrix subspace of dimension* n. *Let* $r = \sqrt{n}$ *and*

$$s = O(\sqrt{n}).$$

Then with at least probability of $1 - \frac{1}{q^r}$, $\mathrm{Ker}_{L,R}(\mathcal{A})$ *consists of matrices of rank* $\leq r$ *for uniformly randomly sampled* $L \in \mathrm{M}(s \times n, \mathbb{F}_q)$ *and* $R \in \mathrm{M}(n \times s, \mathbb{F}_q)$.

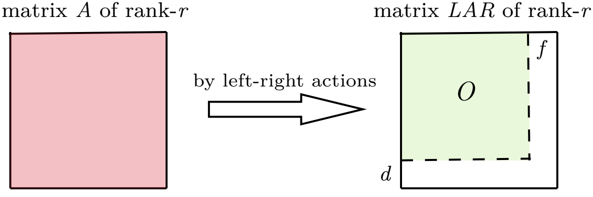
Why is this an advantage?

A trivial case: characterize a low-rank matrix



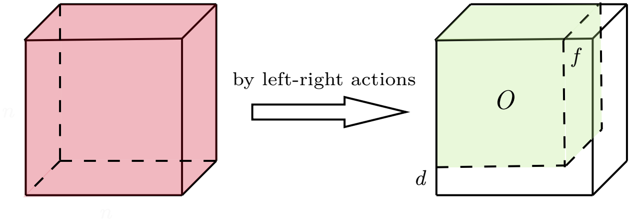
$d + f = O(r)$

A trivial case: characterize a low-rank matrix



$d + f = O(r)$

Technique 2: characterize a low-rank matrix subspace



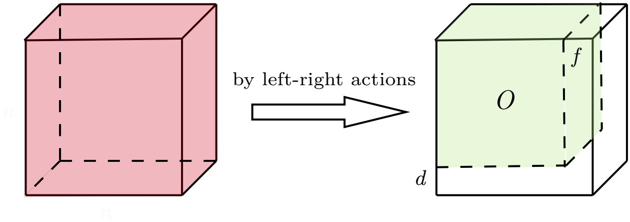
$$d + f = O(r)$$

over field of order $\geq r+1$

[Flanders'62]

Technique 2: characterize a low-rank matrix subspace

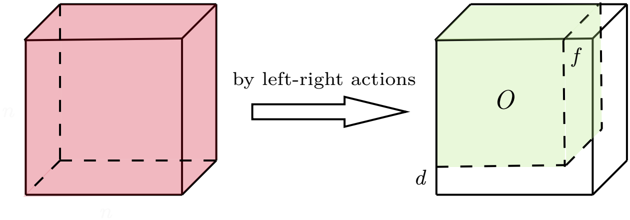
3-tensor bounded by a low rank r Equivalent 3-tensor after characterization



$$d + f = O(r^2)$$

[Sun'23]

Technique 2: characterize a low-rank matrix subspace



3-tensor bounded by a low rank r

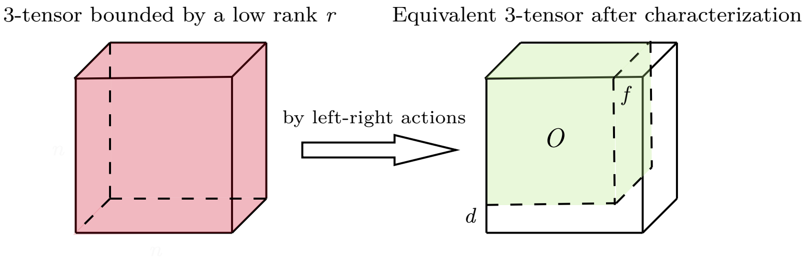
by left-right actions

Equivalent 3-tensor after characterization

$$d + f = O(r \log r)$$

[Ivanyos-Mendoza-Qiao-Sun-Zhang'24]

Technique 2: characterize a low-rank matrix subspace

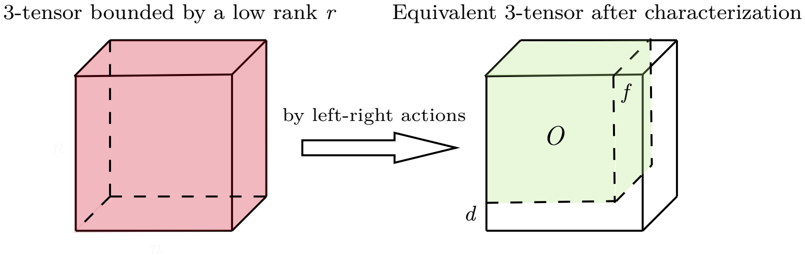


$$d + f = O(r \log r)$$

[Ivanyos-Mendoza-Qiao-Sun-Zhang'24]

- By [Ivanyos-Qiao-Subrahmanyam'18], we can canonicalize the zero block in this case.
- $O(r \log r)$ is obtained from our proof of an inequality between the maximal rank and the non-commutative rank in matrix spaces.

Technique 2: characterize a low-rank matrix subspace

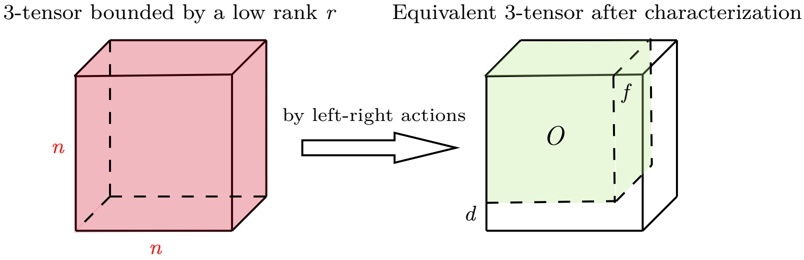


$$d + f = O(r \log r)$$

[Ivanyos-Mendoza-Qiao-Sun-Zhang'24]

- By [Ivanyos-Qiao-Subrahmanyam'18], we can canonicalize the zero block in this case.
- $O(r \log r)$ is obtained from our proof of an inequality between the maximal rank and the non-commutative rank in matrix spaces.

Technique 2: characterize a low-rank matrix subspace

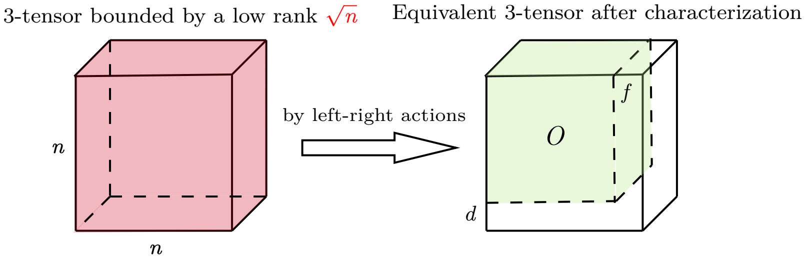


$$d + f = O(r \log r)$$

[Ivanyos-Mendoza-Qiao-Sun-Zhang'24]

- By [Ivanyos-Qiao-Subrahmanyam'18], we can canonicalize the zero block in this case.
- $O(r \log r)$ is obtained from our proof of an inequality between the maximal rank and the non-commutative rank in matrix spaces.

Technique 2: characterize a low-rank matrix subspace

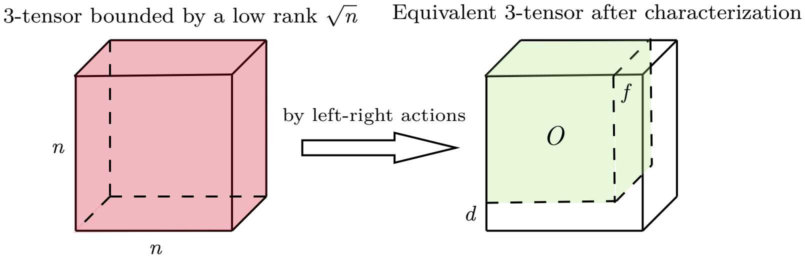


$$d + f = O(\sqrt{n} \log n)$$

[Ivanyos-Mendoza-Qiao-Sun-Zhang'24]

- By [Ivanyos-Qiao-Subrahmanyam'18], we can canonicalize the zero block in this case.
- $O(r \log r)$ is obtained from our proof of an inequality between the maximal rank and the non-commutative rank in matrix spaces.

Technique 2: characterize a low-rank matrix subspace

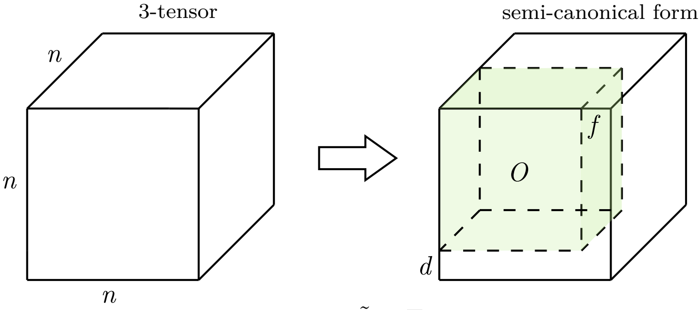


$$d + f = \tilde{O}(\sqrt{n})$$

[Ivanyos-Mendoza-Qiao-Sun-Zhang'24]

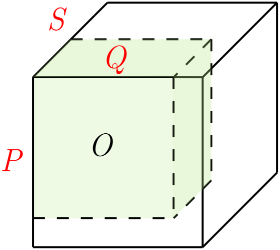
- By [Ivanyos-Qiao-Subrahmanyam'18], we can canonicalize the zero block in this case.
- $O(r \log r)$ is obtained from our proof of an inequality between the maximal rank and the non-commutative rank in matrix spaces.

From semi-canonical 3-tensors to matrix tuples

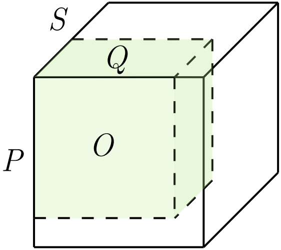


$$d + f = \tilde{O}(\sqrt{n})$$

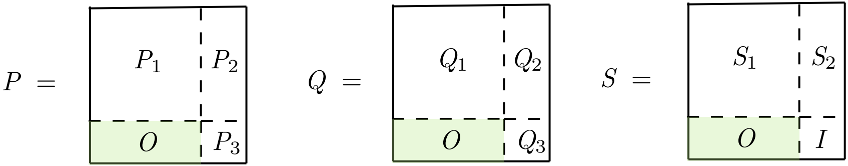
From semi-canonical 3-tensors to matrix tuples



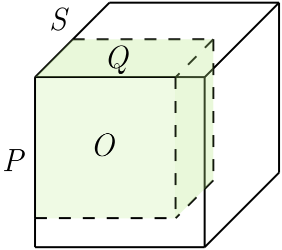
From semi-canonical 3-tensors to matrix tuples



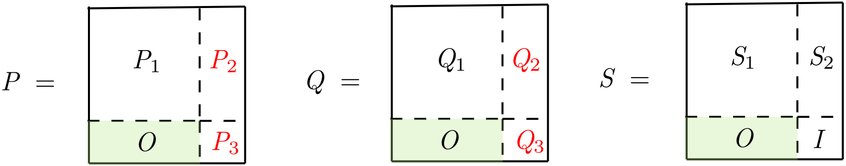
Upon enumeration which costs $q^{\tilde{O}(n^{1.5})}$,



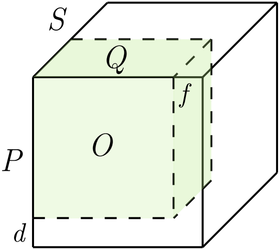
From semi-canonical 3-tensors to matrix tuples



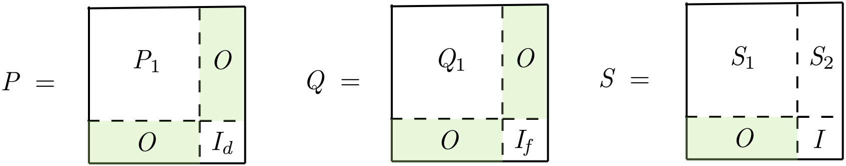
Upon enumeration which costs $q^{\tilde{O}(n^{1.5})}$,



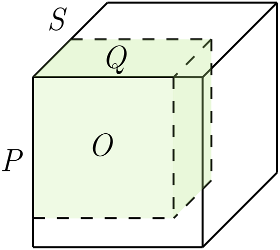
From semi-canonical 3-tensors to matrix tuples



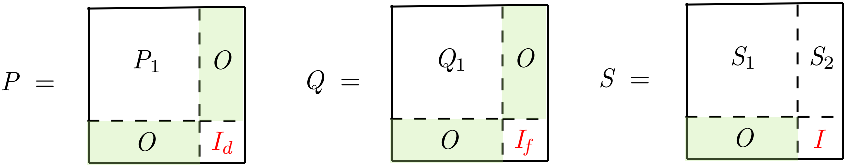
Upon enumeration which costs $q^{\tilde{O}(n^{1.5})}$,



From semi-canonical 3-tensors to matrix tuples

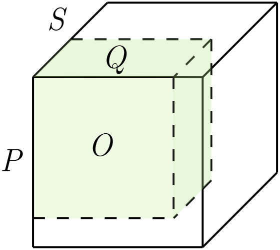


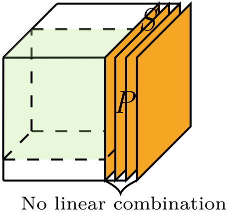
Upon enumeration which costs $q^{\tilde{O}(n^{1.5})}$,

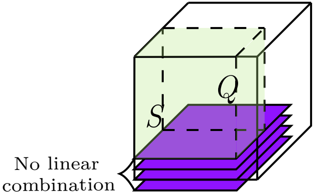


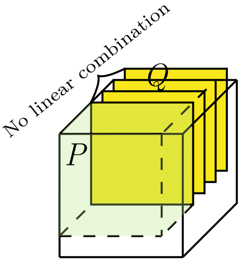
$$P = \begin{pmatrix} P_1 & O \\ O & I_d \end{pmatrix} \qquad Q = \begin{pmatrix} Q_1 & O \\ O & I_f \end{pmatrix} \qquad S = \begin{pmatrix} S_1 & S_2 \\ O & I \end{pmatrix}$$

From semi-canonical 3-tensors to matrix tuples

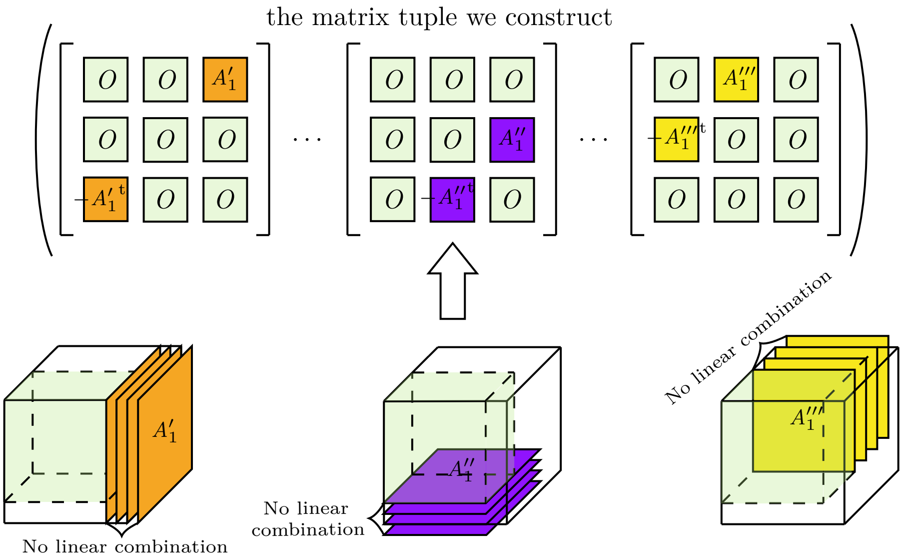






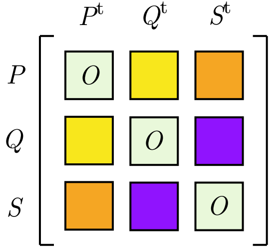


From semi-canonical 3-tensors to matrix tuples

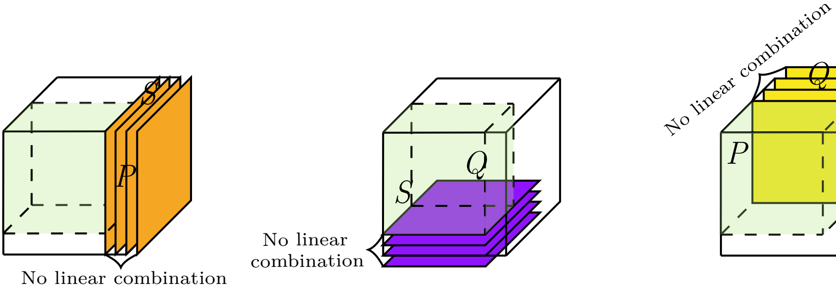


Some colourful slices may be transposed appropriately to match the action matrices.

From semi-canonical 3-tensors to matrix tuples

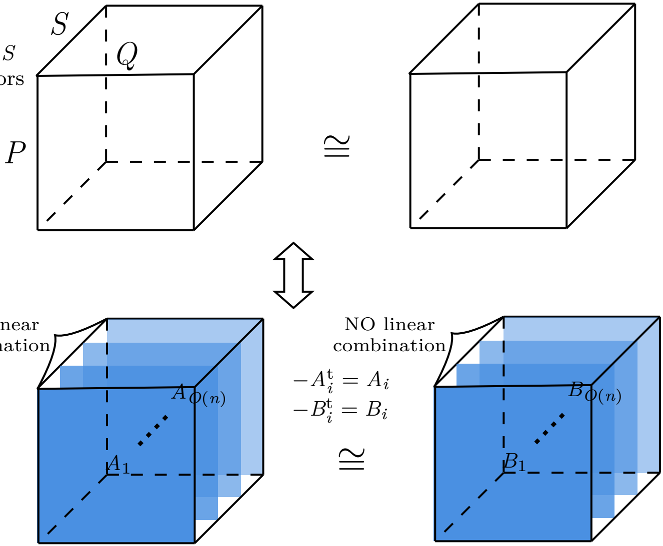


$$T = \begin{bmatrix} P & & \\ & Q & \\ & & S \end{bmatrix}$$



Some colourful slices may be transposed appropriately to match the action matrices.

From 3-TENSOR ISO to (skew-symmetric) TUPLE ISO



∃ invertible matrices P, Q, S s.t. two 3-tensors $\cong$

$\Updownarrow$

$$-A_i^{\mathrm{t}} = A_i$$
$$-B_i^{\mathrm{t}} = B_i$$

$\cong$

∃ an invertible matrix T s.t. $(T^{\mathrm{t}} A_1 T, \cdots, T^{\mathrm{t}} A_{O(n)} T) = (B_1, \cdots, B_{O(n)})$

T is conditioned in a special form, but it is still reducible to the general problem.

Wrap-up of all the results

Theorem (Ivanyos-Qiao'19)

Given two skew-symmetric matrix tuples over $\mathbb{F}_q$, there exists a polynomial-time algorithm that decides whether they are congruent.

Theorem (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Given two $n \times n \times n$ tensors over $\mathbb{F}_q$, there exists an algorithm in time $q^{\tilde{O}(n^{1.5})}$ that decides whether they are equivalent.

Theorem (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Given two p-groups of Frattini class-2 of order N, there exists an algorithm in time $N^{\tilde{O}((\log N)^{1/2})}$ to decide whether they are isomorphic.

Wrap-up of all the results

Theorem (Ivanyos-Qiao'19)

Given two skew-symmetric matrix tuples over $\mathbb{F}_q$*, there exists a polynomial-time algorithm that decides whether they are congruent.*

Theorem (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Given two $n \times n \times n$ *tensors over* $\mathbb{F}_q$*, there exists an algorithm in time* $q^{\tilde{O}(n^{1.5})}$ *that decides whether they are equivalent.*

Theorem (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Given two p-groups of Frattini class-2 of order N*, there exists an algorithm in time* $N^{\tilde{O}((\log N)^{1/2})}$ *to decide whether they are isomorphic.*

Wrap-up of all the results

Theorem (Ivanyos-Qiao'19)

Given two skew-symmetric matrix tuples over $\mathbb{F}_q$*, there exists a polynomial-time algorithm that decides whether they are congruent.*

Theorem (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Given two $n \times n \times n$ *tensors over* $\mathbb{F}_q$*, there exists an algorithm in time* $q^{\tilde{O}(n^{1.5})}$ *that decides whether they are equivalent.*

Theorem (Ivanyos-Mendoza-Qiao-Sun-Zhang'24)

Given two p*-groups of Frattini class-2 of order* N*, there exists an algorithm in time* $N^{\tilde{O}((\log N)^{1/2})}$ *to decide whether they are isomorphic.*

Further directions

- Can we design a similar algorithm for 4-Tensor Iso problem?
- Can we design more faster practical algorithms to break isomorphism-based cryptography protocols?
- Beyond isomorphism testing, 3-tensors themselves are intriguing objects.

Further directions

- Can we design a similar algorithm for 4-TENSOR ISO problem?
- Can we design more faster practical algorithms to break isomorphism-based cryptography protocols?
  - [Narayanan-Qiao-Tang'24] made a heuristic one running in time $q^{n/2}$ for the average case of the equivalence testing of 3-tensors.
- Beyond isomorphism testing, 3-tensors themselves are intriguing objects.

Further directions

- Can we design a similar algorithm for 4-Tensor Iso problem?
- Can we design more faster practical algorithms to break isomorphism-based cryptography protocols?
  - [Narayanan-Qiao-Tang'24] made a heuristic one running in time $q^{n/2}$ for the average case of the equivalence testing of 3-tensors.
- Beyond isomorphism testing, 3-tensors themselves are intriguing objects.

Further directions

- Can we design a similar algorithm for 4-TENSOR ISO problem?
- Can we design more faster practical algorithms to break isomorphism-based cryptography protocols?
  - [Narayanan-Qiao-Tang'24] made a heuristic one running in time $q^{n/2}$ for the average case of the equivalence testing of 3-tensors.
- Beyond isomorphism testing, 3-tensors themselves are intriguing objects. If anyone is interested, we can talk offline about two of my previous papers on the connections between properties of graphs and linear spaces of matrices [Li-Qiao-Wigderson-Wigderson-Zhang'22&23].

Further directions

- Can we design a similar algorithm for 4-TENSOR ISO problem?
- Can we design more faster practical algorithms to break isomorphism-based cryptography protocols?
  - [Narayanan-Qiao-Tang'24] made a heuristic one running in time $q^{n/2}$ for the average case of the equivalence testing of 3-tensors.
- Beyond isomorphism testing, 3-tensors themselves are intriguing objects. If anyone is interested, we can talk offline about two of my previous papers on the connections between properties of graphs and linear spaces of matrices [Li-Qiao-Wigderson-Wigderson-Zhang'22&23].

Question and Answer

Thank you so much!

Please find the paper and slides available on my webpage:

https://www.chuanqizhang.com