Some algebraic algorithms and complexity classes inspired by connections between matrix spaces and graphs

Youming Qiao
University of Technology Sydney

@ Workshop on Algebraic Complexity Theory (WACT) 2023
30 March, 2023

Some typos corrected, N.B. added,
on 5 April.

Talk outline

1. Some connections between graphs and matrix spaces
2. Algorithm: alternating paths and Wong sequences
3. Complexity: graph isomorphism and matrix space equivalence
4. More connections, more problems

* Based on the following joint works:

- Yinan Li, Youming Qiao, Avi Wigderson, Yuval Wigderson, Chuanqi Zhang: Connections between graphs and matrix spaces. CoRR abs/2206.04815 (2022). To appear in Israel J Maths
- Joshua A. Grochow, Youming Qiao: On the complexity of isomorphism problems for tensors, groups, and polynomials I: Tensor Isomorphism-completeness. ITCS 2021: 31:1-31:19.
- Gábor Ivanyos, Youming Qiao, K. V. Subrahmanyam: Constructive non-commutative rank computation is in deterministic polynomial time. Comput. Complex. 27(4): 561-593 (2018).
- Yinan Li, Youming Qiao: Linear algebraic analogues of the graph isomorphism problem and the Erdős-Rényi model. FOCS 2017: 463-474.

From graphs to matrix spaces

* For $n \in \mathbb{N}$, $[n] := \{1, 2, \cdots, n\}$. $\mathbb{F}$: a field

* $M(n, \mathbb{F})$: the linear space of $n \times n$ matrices over $\mathbb{F}$

* For $i, j \in [n]$, $E_{i,j} \in M(n, \mathbb{F})$ is the (i,j)th elementary matrix

$$E_{i,j} = \begin{array}{cc} & \begin{array}{cccc} & & j & \end{array} \\ i & \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & 1 & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \end{array}$$

From graphs to matrix spaces

* For $n \in \mathbb{N}$, $[n] := \{1, 2, \cdots, n\}$. $\mathbb{F}$: a field

* $M(n, \mathbb{F})$: the linear space of $n \times n$ matrices over $\mathbb{F}$

* For $i, j \in [n]$, $E_{i,j} \in M(n, \mathbb{F})$ is the (i,j)th elementary matrix

* A bipartite graph $G = (L \cup R, F)$ $\Rightarrow$ A matrix space $B_G \leq M(n, \mathbb{F})$
  $L = R = [n]$, $F \subseteq L \times R$ $\quad$ $B_G = \text{span}\{E_{i,j} \mid (i,j) \in F\}$

From graphs to matrix spaces

* For $n \in \mathbb{N}$, $[n] := \{1, 2, \cdots, n\}$. $\mathbb{F}$: a field

* $M(n, \mathbb{F})$: the linear space of $n \times n$ matrices over $\mathbb{F}$

* For $i, j \in [n]$, $E_{i,j} \in M(n, \mathbb{F})$ is the (i,j)th elementary matrix

* A bipartite graph $G = (L \cup R, F)$ $\Rightarrow$ A matrix space $\mathcal{B}_G \leq M(n, \mathbb{F})$
  $L = R = [n]$, $F \subseteq L \times R$ $\qquad$ $\mathcal{B}_G = \operatorname{span}\{E_{i,j} \mid (i,j) \in F\}$

Observation. (Tutte, Edmonds, Lovász)

G has a perfect matching $\Leftrightarrow$ $\mathcal{B}_G$ contains a full-rank matrix

Connections between graphs and matrix spaces

Observation. (Tutte, Edmonds, Lovász)

G has a perfect matching $\Leftrightarrow$ B_G contains a full-rank matrix

* A classical result of the type: G has property P iff B_G has property Q

Connections between graphs and matrix spaces

Observation. (Tutte, Edmonds, Lovász)

G has a perfect matching $\Leftrightarrow$ B_G contains a full-rank matrix

* A classical result of the type: G has property P iff B_G has property Q

* Symbolic determinant identity testing (SDIT) essentially asks to test if a general matrix space contains a full-rank matrix: a problem of key importance in algebraic complexity [Kabanets-Impagliazzo]

* Quasi-NC algorithm for perfect matching [Fenner-Gurjar-Thierauf]

* We now examine another side of the above observation

Another correspondence between graph and matrix space structures

* $G = (L \cup R, F)$ $\Rightarrow$ $\mathcal{B}_G = \mathrm{span}\{E_{i,j} \mid (i,j) \in F\} \subseteq M(n, \mathbb{F})$
  $L = R = [n]$

Obs. G has a perfect matching $\Leftrightarrow$ $\mathcal{B}_G$ contains a full-rank matrix

Prop. (Hall) G has a shrunk subset $\Leftrightarrow$ $\mathcal{B}_G$ has a shrunk subspace

$S \subseteq L$, $|S| > |N(S)|$
$N(S) \subseteq R$ is the set of neighbours of S

$S \subseteq \mathbb{F}^n$, $\dim(S) > \dim(\mathcal{B}_G(S))$
$\mathcal{B}_G(S) = \mathrm{span}\left(\bigcup_{B \in \mathcal{B}_G} B(S)\right)$

Another correspondence between graph and matrix space structures

* $G = (L \cup R, F)$ $\Rightarrow$ $\mathcal{B}_G = \mathrm{span}\{E_{i,j} \mid (i,j) \in F\} \leq M(n, \mathbb{F})$

  $L = R = [n]$

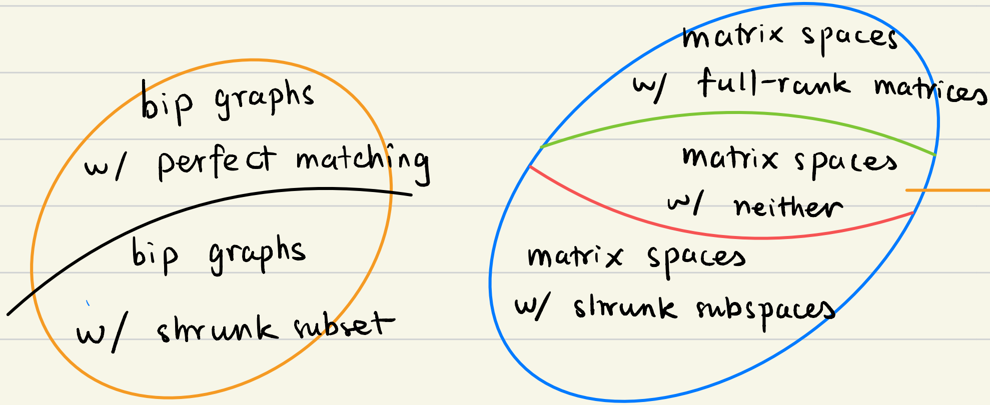
Obs. G has a perfect matching $\Leftrightarrow$ $\mathcal{B}_G$ contains a full-rank matrix

Prop. (Hall) G has a shrunk subset $\Leftrightarrow$ $\mathcal{B}_G$ has a shrunk subspace

* Non-commutative rational identity testing (NC-RIT) essentially asks to test if a general matrix space admits a shrunk subspace [Hrubeš–Wigderson]

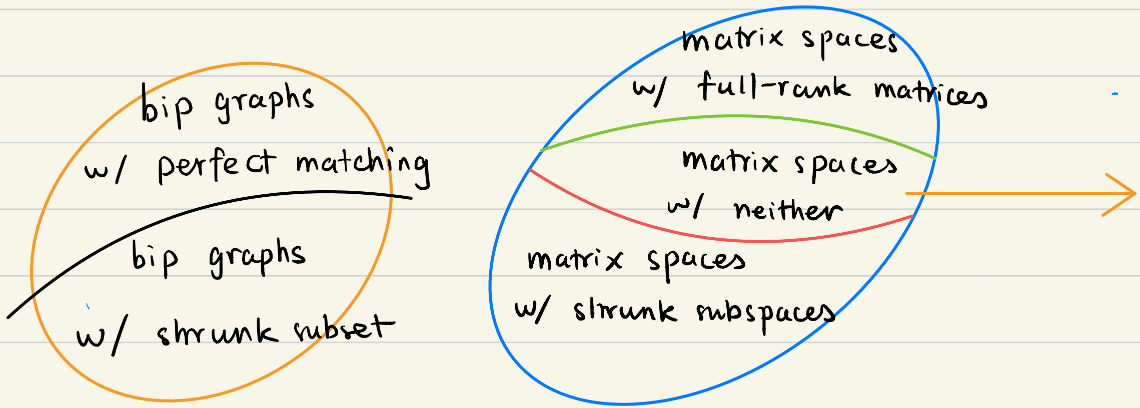
* Geometric complexity theory [Mulmuley], polynomial identity testing [Derksen–Makam], non-commutative algebra [Cohn], analysis [Garg-Gurvits-Oliveira-Wigderson]...

SDIT versus NC-RIT



$$sk_3 = \text{span}\left\{ \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \right\}$$

SDIT versus NC-RIT



$$sk_3 = \text{span}\left\{ \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \right\}$$

* SDIT: in coRP over large fields. A major open problem to derandomise it.

* NC-RIT: in P by [Garg-Gurvits-Oliveira-Wigderson], [Ivanyos-Q-Subrahmanyam], [Hadama-Hirai]

Linear algebraic alternating path method

* The Ivanyos-Q-Subrahmanyam algorithm for NC-RIT:
  - A linear algebraic alternating path method [Ivanyos-Karpinski-Q-Santha]
  - A "regularity lemma" for matrix space blow-ups (via division algebras)

* Alternating path method on bipartite graphs:

* $G = (L \cup R, E)$, $M \subseteq E$ is a given matching, $U = E \setminus M$: edges not in M

$S_0 \subseteq L$: unmatched vertices $\overset{U}{\Rrightarrow}$ $T_1 \subseteq R$: neighbours of S_0 via unmatched edges

- if T_1 contains an unmatched vertex, an augmenting path is found
- otherwise ...

Review of alternating paths on bipartite graphs

* $G = (L \cup R, E)$, $M \subseteq E$ is a given matching, $U = E \setminus M$: edges not in M

$S_0 \subseteq L$: unmatched vertices $\overset{U}{\Longrightarrow}$ $T_1 \subseteq R$: neighbours of S_0 via unmatched edges

$S_1 \subseteq L$: n.b. of T_1 via matched edges $\overset{M}{\Longleftarrow}$

Review of alternating paths on bipartite graphs

* $G = (L \cup R, E)$, $M \subseteq E$ is a given matching, $U = E \setminus M$: edges not in M

$S_0 \subseteq L$: unmatched vertices $\overset{U}{\Longrightarrow}$ $T_1 \subseteq R$: neighbours of S_0 via unmatched edges

$\overset{M}{\Longleftarrow}$

$S_1 \subseteq L$: n.b. of T_1 via matched edges $\overset{U}{\Longrightarrow}$ $T_2 \subseteq R$: n.b. of S_1 via unmatched edges

- Check if T_2 contains an unmatched vertex
- Yes : augmenting path. No : continue

Review of alternating paths on bipartite graphs

* $G = (L \cup R, E)$, $M \subseteq E$ is a given matching, $U = E \setminus M$: edges not in M

$S_0 \subseteq L$: unmatched vertices $\overset{U}{\Longrightarrow}$ $T_1 \subseteq R$: neighbours of S_0 via unmatched edges

$\overset{M}{\Longleftarrow}$

$S_1 \subseteq L$: n.b. of T_1 via matched edges

$\overset{U}{\Longrightarrow}$ $T_2 \subseteq R$: n.b. of S_1 via unmatched edges

- Check if T_2 contains an unmatched vertex
- Yes : augmenting path. No : continue

⋮

STOP if T_i consists of matched vertices and $T_i \subseteq T_1 \cup T_2 \cup \cdots \cup T_{i-1}$

Linear algebraic alternating path method

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$. $C \in \mathcal{B}$

$\underline{S_0} = \ker(C) \subseteq \mathbb{F}^n$

"unmatched vertices"

Linear algebraic alternating path method

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$. $C \in \mathcal{B}$

"neighbors of S_0 via unmatched edges"

$$S_0 = \ker(C) \subseteq \mathbb{F}^n \xRightarrow{\mathcal{B}} T_1 = \mathcal{B}(S_0) := \mathrm{span}\{B_1(S_0) \cup \cdots \cup B_m(S_0)\} \subseteq \mathbb{F}^n$$

Linear algebraic alternating path method

* $\mathcal{B} = \text{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$. $C \in \mathcal{B}$

$S_0 = \ker(C) \subseteq \mathbb{F}^n \overset{\mathcal{B}}{\Longrightarrow} T_1 = \mathcal{B}(S_0) := \text{span}\{B_1(S_0) \cup \cdots \cup B_m(S_0)\} \subseteq \mathbb{F}^n$

- If $\underbrace{T_1 \nsubseteq \text{im}(C)}$, can compute $D \in \mathcal{B}$ of larger rank
- Otherwise ... "T_1 contains an unmatched vector"

Linear algebraic alternating path method

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$. $C \in \mathcal{B}$

$$S_0 = \ker(C) \subseteq \mathbb{F}^n \xRightarrow{\mathcal{B}} T_1 = \mathcal{B}(S_0) := \mathrm{span}\{B_1(S_0) \cup \cdots \cup B_m(S_0)\} \subseteq \mathrm{Im}(C)$$

$$\xLeftarrow{C^{-1}}$$

$$S_1 = C^{-1}(T_1) := \{v \in \mathbb{F}^n \mid C(v) \in T_1\}$$

Linear algebraic alternating path method

* $\mathcal{B} = \text{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$. $C \in \mathcal{B}$

$$S_0 = \ker(C) \subseteq \mathbb{F}^n \xRightarrow{\mathcal{B}} T_1 = \mathcal{B}(S_0) := \text{span}\{B_1(S_0) \cup \cdots \cup B_m(S_0)\} \subseteq \text{Im}(C)$$

$$\xLeftarrow{C^{-1}}$$

$$S_1 = C^{-1}(T_1) := \{v \in \mathbb{F}^n \mid C(v) \in T_1\}$$

$$\xRightarrow{\mathcal{B}} T_2 = \mathcal{B}(S_1)$$

- Check if $T_2 \nsubseteq \text{im}(C)$.
- Yes: cannot find D of larger rank in $\mathcal{B}$
  but "do so in $\mathcal{B} \otimes M(n, \mathbb{F})$"
- No: continue

Linear algebraic alternating path method

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$. $C \in \mathcal{B}$

$$S_0 = \ker(C) \subseteq \mathbb{F}^n \xRightarrow{\mathcal{B}} T_1 = \mathcal{B}(S_0) := \mathrm{span}\{B_1(S_0) \cup \cdots \cup B_m(S_0)\} \subseteq \mathbb{F}^n$$

$$\xLeftarrow{C^{-1}}$$

$$S_1 = C^{-1}(T_1) := \{v \in \mathbb{F}^n \mid C(v) \in T_1\}$$

$$\xRightarrow{\mathcal{B}} T_2 = \mathcal{B}(S_1)$$

$$\vdots \quad \text{STOP if } T_{i+1} = T_i \leq \mathrm{im}(C)$$

Lemma. [Ivanyos-Karpinski-Q-Santha] $\mathcal{B}$ has a shrunk subspace of gap $\mathrm{corank}(C)$ iff $\exists i$, $T_{i+1} = T_i \leq \mathrm{im}(C)$

Recap for the NC-RIT story

* Start with "G has property P iff B_G has property Q"

* Go on to examine the problem of testing "B has property Q"

N.B. This is just one way of arriving at NC-RIT

Recap for the NC-RIT story

* Start with “G has property P iff B_G has property Q”

* Go on to examine the problem of testing “B has property Q”

* Inspired by techniques for solving the problem of testing “G has property P”

    1. [Garg-Gurvits-Oliveira-Wigderson] Sinkhorn's scaling algorithm
    2. [Ivanyos-Q-Subrahmanyam] the augmenting path algorithm
    3. [Hamada-Hirai] submodular optimisation

Recap for the NC-RIT story

* Start with "G has property P iff B_G has property Q"

* Go on to examine the problem of testing "B has property Q"

* Inspired by techniques for solving the problem of testing "G has property P"

  1. [Garg-Gurvits-Oliveira-Wigderson] Sinkhorn's scaling algorithm
  2. [Ivanyos-Q-Subrahmanyam] the augmenting path algorithm
  3. [Hamada-Hirai] submodular optimisation

* The situation is usually more complicated for testing "B has property Q"

  - The discrepancy between "full-rank matrices" and "shrunk subspaces"

    not having

Graph isomorphism versus matrix space equivalence

Def. $G_1 = (L \cup R, F_1)$ and $G_2 = (L \cup R, F_2)$, $L = R = [n]$, $F_1, F_2 \subseteq L \times R$ are isomorphic, if $\exists\, \sigma, \pi \in S_n$, such that $(i,j) \in F_1 \Leftrightarrow (\sigma(i), \pi(j)) \in F_2$

* Bipartite graph iso is as hard as general graph iso

Graph isomorphism versus matrix space equivalence

Def. $G_1 = (L \cup R, F_1)$ and $G_2 = (L \cup R, F_2)$, $L = R = [n]$, $F_1, F_2 \subseteq L \times R$ are isomorphic, if $\exists\, \sigma, \pi \in S_n$, such that $(i, j) \in F_1 \Leftrightarrow (\sigma(i), \pi(j)) \in F_2$

* $A_1, A_2 \in M(n, \mathbb{F})$ are equivalent, if $\exists\, L, R \in GL(n, \mathbb{F})$, $A_1 = L A_2 R$

Def. Matrix spaces $\mathcal{B}_1, \mathcal{B}_2 \leq M(n, \mathbb{F})$ are equivalent, if $\exists\, L, R \in GL(n, \mathbb{F})$ such that $\mathcal{B}_1 = L \mathcal{B}_2 R := \{ LBR \mid B \in \mathcal{B}_2 \}$

Graph isomorphism versus matrix space equivalence

Def. $G_1 = (L \cup R, F_1)$ and $G_2 = (L \cup R, F_2)$, $L = R = [n]$, $F_1, F_2 \subseteq L \times R$ are isomorphic, if $\exists\, \sigma, \pi \in S_n$, such that $(i,j) \in F_1 \iff (\sigma(i), \pi(j)) \in F_2$

* $A_1, A_2 \in M(n, \mathbb{F})$ are equivalent, if $\exists\, L, R \in GL(n, \mathbb{F})$, $A_1 = L A_2 R$

Def. Matrix spaces $\mathcal{B}_1, \mathcal{B}_2 \leq M(n, \mathbb{F})$ are equivalent, if $\exists\, L, R \in GL(n, \mathbb{F})$ such that $\mathcal{B}_1 = L \mathcal{B}_2 R := \{ LBR \mid B \in \mathcal{B}_2 \}$

Prop. [Li-Q-Wigderson-Wigderson-Zhang] N.B. adapted from Prop 6.2 there

G and H are isomorphic $\iff$ B_G and B_H are equivalent

Matrix space equivalence

N.B. This gives a poly-time reduction from GraphIso to TensorIso

Prop. [Li-Q-Wigderson-Wigderson-Zhang]

G and H are isomorphic $\Leftrightarrow$ B_G and B_H are equivalent

* Matrix space equivalence as a proper generalisation of graph isomorphism

* Next step: matrix space equivalence for general matrix spaces

Matrix space equivalence

Prop. [Li-Q-Wigderson-Wigderson-Zhang]

G and H are isomorphic $\Leftrightarrow$ B_G and B_H are equivalent

* Matrix space equivalence as a proper generalisation of graph isomorphism

* Next step: matrix space equivalence for general matrix spaces

* Results inspired by the study of graph isomorphism?
  - [Li-Q]: individualisation and refinement as used in [Babai-Erdős-Selkow]

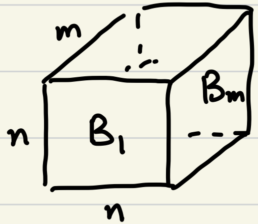
* [Grochow-Q]: a complexity class called Tensor Isomorphism (TI) in analogy with GI
  - A gadget design in analogy with some method from colored graph isomorphism

Matrix space equivalence as tensor isomorphism

* [Grochow-Q]: a complexity class called Tensor Isomorphism (TI) in analogy with GI

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$

$\Downarrow$

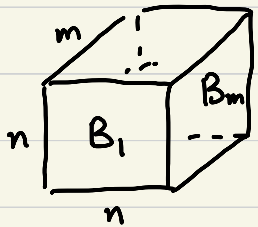
$T_{\mathcal{B}} =$  $\in \mathbb{F}^n \otimes \mathbb{F}^n \otimes \mathbb{F}^m$

Matrix space equivalence as tensor isomorphism

* [Grochow-Q]: a complexity class called Tensor Isomorphism (TI) in analogy with GI

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$

$\Downarrow$

$T_{\mathcal{B}} =$ 

$\in \mathbb{F}^n \otimes \mathbb{F}^n \otimes \mathbb{F}^m$

$\mathcal{B}, \mathcal{C} \subseteq M(n, \mathbb{F})$ are equivalent

$\Updownarrow$

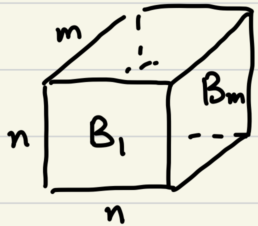
$T_{\mathcal{B}}, T_{\mathcal{C}}$ are isomorphic, i.e. in the same orbit under $GL(n, \mathbb{F}) \times GL(n, \mathbb{F}) \times GL(m, \mathbb{F})$.

Matrix space equivalence as tensor isomorphism

* $\mathcal{B} = \mathrm{span}\{B_1, \cdots, B_m\} \subseteq M(n, \mathbb{F})$

$\Downarrow$

$T_{\mathcal{B}} =$



$\in \mathbb{F}^n \otimes \mathbb{F}^n \otimes \mathbb{F}^m$

$\mathcal{B}, \mathcal{C} \subseteq M(n, \mathbb{F})$ are equivalent

$\Updownarrow$

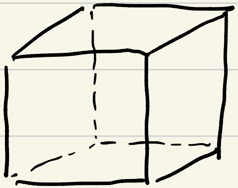
$T_{\mathcal{B}}, T_{\mathcal{C}}$ are isomorphic, i.e. in the same orbit under $GL(n, \mathbb{F}) \times GL(n, \mathbb{F}) \times GL(m, \mathbb{F})$.

Def. [Grochow-Q] The complexity class TI consists of problems polynomial-time reducible to the matrix space equivalence = 3-tensor isomorphism problem.

* Wishful thinking: just as GI captures isomorphism problems for combinatorial structures, TI captures isomorphism problems for algebraic structures

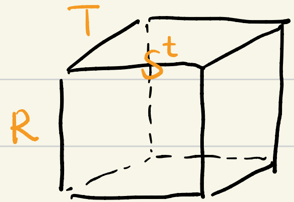
Actions on 3-way arrays

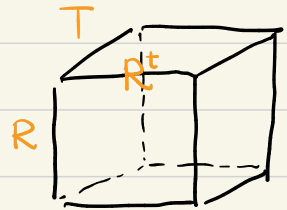
* $R, S, T \in GL(n, \mathbb{F})$

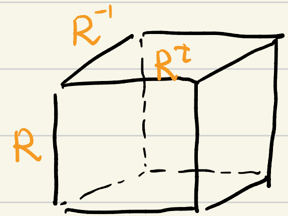


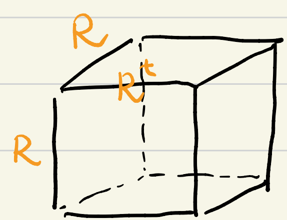
$A = (a_{i,j,k})$

$i, j, k \in [n]$



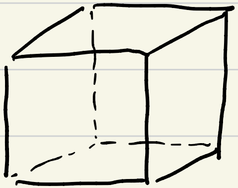






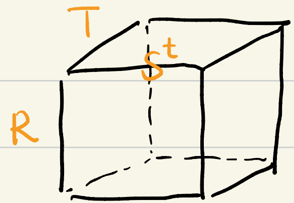
Actions on 3-way arrays

* $R, S, T \in GL(n, \mathbb{F})$



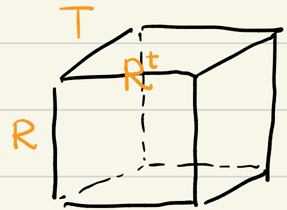
$A = (a_{i,j,k})$
$i, j, k \in [n]$

* $\mathcal{U}, V, W \cong \mathbb{F}^n$



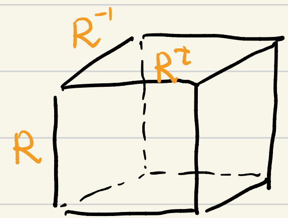
Tensor

$t: \mathcal{U} \times V \times W \to \mathbb{F}$



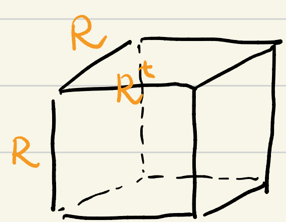
Bilinear map

$f: \mathcal{U} \times \mathcal{U} \to W$



Algebra

$a: \mathcal{U} \times \mathcal{U} \to \mathcal{U}$



Trilinear Form

$c: \mathcal{U} \times \mathcal{U} \times \mathcal{U} \to \mathbb{F}$

3-way arrays are versatile

* Under different actions, 3-way arrays encode tensors, bilinear maps, algebras, and trilinear forms

* Putting some structural restrictions we get more

1. Symmetric bilinear maps $f: U \times U \to V$: systems of quadratic forms

2. Skew-symmetric bilinear maps over $GF(p)$: p-groups of class 2 and exponent p

3. Symmetric trilinear forms over F, $char(F)$ not 2 or 3: cubic forms

4. Associativity, Jacobi conditions…: associative algebras or Lie algebras

TI-complete problems

Theorem. [Futorny-Grochow-Sergeichuk, Grochow-Q]
The following problems are TI-complete:

- Isomorphism of p-groups of class 2 and exponent p, given by matrix groups
- Isomorphism of systems of quadratic forms, cubic forms
- Isomorphism of associative and Lie algebras

TI-complete problems

Theorem. [Futorny-Grochow-Sergeichuk, Grochow-Q]
The following problems are TI-complete:
- Isomorphism of p-groups of class 2 and exponent p, given by matrix groups
- Isomorphism of systems of quadratic forms, cubic forms
- Isomorphism of associative and Lie algebras

* How about d-tensors for d>3? Note that 2-tensor isomorphism (matrix equivalence) is easy.

d, d>3

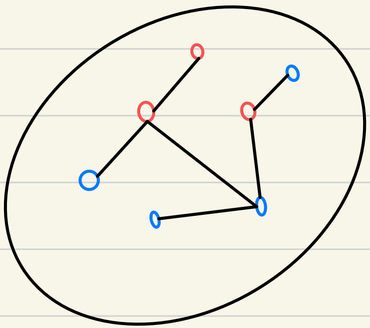
Theorem. [Grochow-Q] ~~k~~-tensor isomorphism reduces to 3-tensor isomorphism.

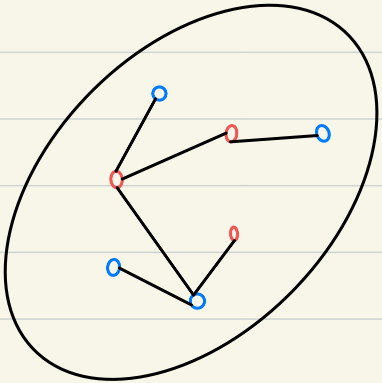
* In the spirit that 3SAT is NP-complete, and 2SAT is in P.

Methods for relating the problems

* Two techniques for relating 3-way arrays under different actions: Gelfand-Panomerav and gadget methods

* The gadgets are reminiscent of those used for colored graph isomorphism

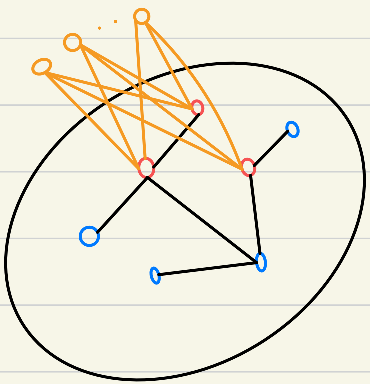


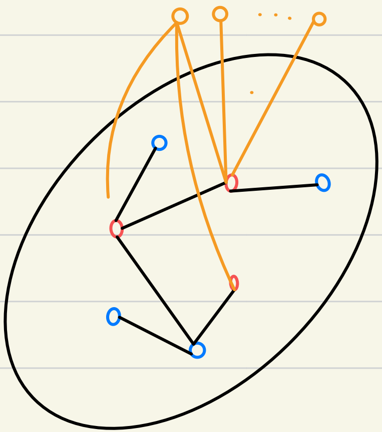


Methods for relating the problems

* Two techniques for relating 3-way arrays under different actions: Gelfand-Panomerav and gadget methods

* The gadgets are reminiscent of those used for colored graph isomorphism





- Star gadgets:
Degrees of red vertices are large enough so blue vertices cannot be mapped to them

One example of the reductions

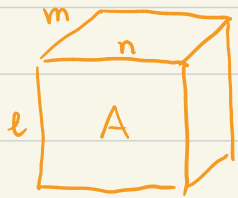
Goal. Given $f, g : U \times V \times W \to \mathbb{F}$, construct $\hat{f}, \hat{g} : S \times S \to T$, skew-symmetric such that $f \sim g$ under $GL(U) \times GL(V) \times GL(W)$ iff $\hat{f} \sim \hat{g}$ under $GL(S) \times GL(T)$

Construction.

$\dim(U) = \ell$

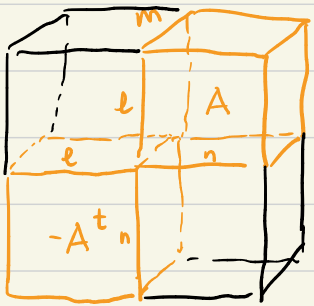
$\dim(V) = n$

$\dim(W) = m$



$\Rightarrow$ $S = U \oplus V$

$T = W$



(Entries outside the orange region are 0).

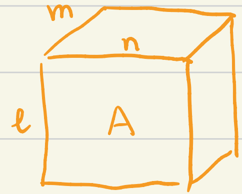
From tensors to bilinear maps

Construction.

$\dim(U) = \ell$

$\dim(V) = n$

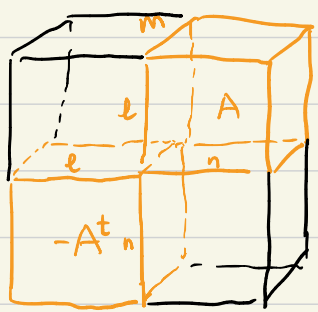
$\dim(W) = m$



$\Downarrow$

$S = U \oplus V$

$T = W$



(Entries outside the orange region are 0).

* This construction does not work because GL(S) may mix U with V. So we need:

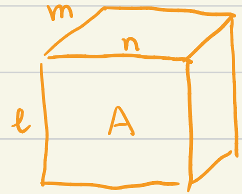
From tensors to bilinear maps

Construction.

$\dim(U) = \ell$

$\dim(V) = n$

$\dim(W) = m$



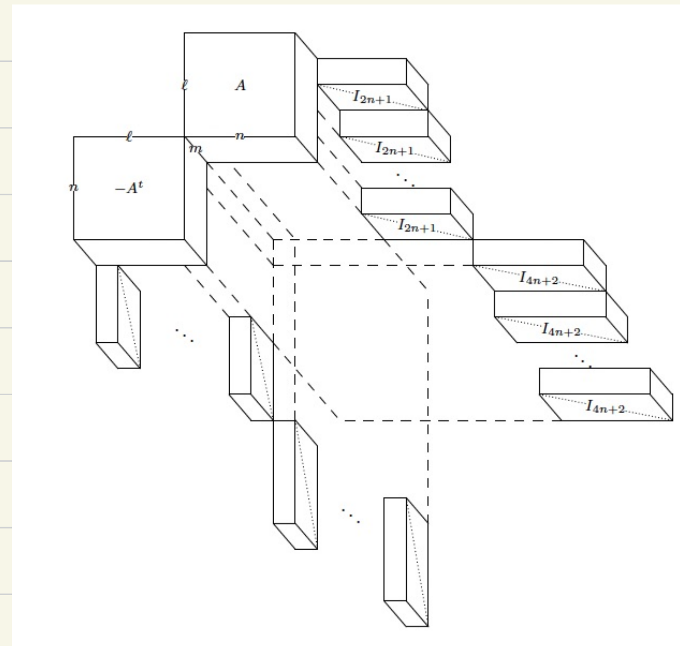
$\Downarrow$

$S = U \oplus V$

$T = W$

(Entries outside the orange region are 0).

* This construction does not work because GL(S) may mix U with V. So we need:



More correspondences, more questions

* A directed graph $G = (V, F)$ $\Rightarrow$ $\mathcal{B}_G = \mathrm{span}\{E_{i,j} \mid (i,j) \in F\} \subseteq M(n, \mathbb{F})$.
  $V = [n]$, $F \subseteq V \times V$

Prop. [Li-Q-Wigderson-Wigderson-Zhang]

G is acyclic $\Leftrightarrow$ $\mathcal{B}_G$ contains only nilpotent matrices

* Not so surprising, but ..

More correspondences, more questions

* A directed graph $G = (V, F)$ $\Rightarrow$ $B_G = \mathrm{span}\{E_{i,j} \mid (i,j) \in F\} \leq M(n, \mathbb{F})$.
  $V = [n]$, $F \subseteq V \times V$

Prop. [Li-Q-Wigderson-Wigderson-Zhang]
G is acyclic $\Leftrightarrow$ B_G contains only nilpotent matrices

Prop. [ibid.] Max size over acyclic subgraphs in G
= Max dim over nilpotent subspaces in B_G

* Generalise Gerstenhaber's result :
  max dim of nilpotent matrix spaces in $M(n, \mathbb{F}) = \binom{n}{2}$

Matrix space nilpotency testing

Def. (Matrix space nilpotency testing) Given a linear basis of a matrix space B, decide if B contains only nilpotent matrices.

* Given a symbolic matrix S of size n, decide if S^n is the zero matrix.

* Reduces to SDIT, which is equivalent to asking whether the $(1, 1)$ entry of S^n is 0

* The naturally associated group action is matrix conjugation (instead of left-right) on matrix tuples. The nullcone problem, rank-1 spanned setting, etc. are easier.

* SDIT reduces to computing the nilpotency index [Li-Q-Wigderson-Wigderson-Zhang]

Brief summary

* A pattern of the stories:

1. Start with "G has property P iff B_G has property Q"
2. Ask the question "B has property Q"
3. Devise linear algebraic analogues of graph-theoretic methods

* Shrunk subset vs shrunk subspace, graph isomorphism vs tensor isomorphism

* Alternating paths vs Wong sequences, graph coloring gadgets vs rank gadgets

* Will matrix space nilpotency test be the next target?

Thank you!

And questions please :)