Constructive Recognition of Classical Groups

supervised by Prof. Alice Niemeyer and Prof. Max Horn

Daniel Rademacher

08.09.2022

Algebra and Representation Theory

RWTH AACHEN UNIVERSITY

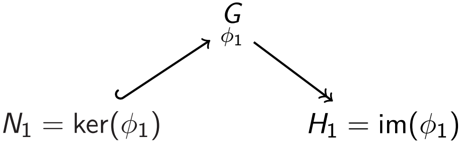
Input: $G \leq \mathrm{GL}(n, q)$.
Output: recognition tree.

Input: $G \leq \mathrm{GL}(n, q)$.
Output: recognition tree.

G

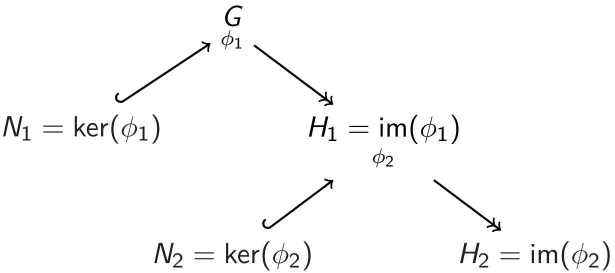
Recognition Tree

Input: $G \leq \mathrm{GL}(n, q)$.
Output: recognition tree.



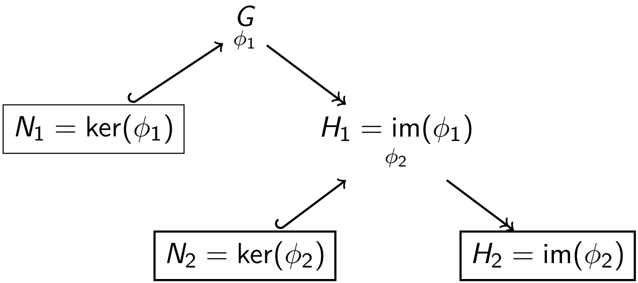
Recognition Tree

Input: $G \leq \mathrm{GL}(n, q)$.
Output: recognition tree.



Recognition Tree

Input: $G \leq \mathrm{GL}(n, q)$.
Output: recognition tree.



Leaves are **(almost) simple groups**.

Contents

Motivation

Special Linear Group

Let $d \in \mathbb{N}$, $q = p^f$ a prime power and define

$$SL(d, q) := \{a \in GL(d, q) \mid \det(a) = 1\}.$$

Word problem

Given $\langle X \rangle = G \leq \mathrm{GL}(d, q)$ and $a \in G$. The problem to write a as a word in the generators X is called the *word problem*.

Transvection

$E_{i,j}(\alpha)$

We define $E_{i,j}(\alpha) \in \mathbb{F}^{d\times d}$ for $i, j \in \{1, \ldots, d\}$ with $i \neq j$ and $\alpha \in \mathbb{F}*$ as follows:

$$(E_{i,j}(\alpha))_{a,b} = \begin{cases} 1, & \text{if } a = b, \\ \alpha, & \text{if } i = a \text{ and } j = b, \\ 0, & \text{else.} \end{cases}$$

The matrices $E_{i,j}(\alpha)$ for $i, j \in \{1, \ldots, d\}$ with $i \neq j$ and $\alpha \in \mathbb{F}^*$ are transvections.

Transvection

$$E_{2,5}(3) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 3 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \in \mathrm{SL}(6, 7)$$

Theorem

$\mathrm{SL}(d, q)$ is generated by transvections.

$$\begin{pmatrix} 5 & \color{red}{0} & 6 & 3 \\ 5 & 5 & 2 & 0 \\ 1 & 4 & 6 & 6 \\ 5 & 5 & 2 & 2 \end{pmatrix} \in \mathrm{SL}(4,7)$$

Example

$$\begin{pmatrix} 5 & {\color{red}0} & 6 & 3 \\ 5 & 5 & 2 & 0 \\ 1 & 4 & 6 & 6 \\ 5 & 5 & 2 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 5 & {\color{green}5} & 6 & 3 \\ 5 & 3 & 2 & 0 \\ 1 & 5 & 6 & 6 \\ 5 & 3 & 2 & 2 \end{pmatrix} \rightarrow$$

Example

$$\begin{pmatrix} 5 & \color{red}{0} & 6 & 3 \\ 5 & 5 & 2 & 0 \\ 1 & 4 & 6 & 6 \\ 5 & 5 & 2 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 5 & \color{green}{5} & 6 & 3 \\ 5 & 3 & 2 & 0 \\ 1 & 5 & 6 & 6 \\ 5 & 3 & 2 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} \color{red}{5} & 5 & 6 & 3 \\ 5 & 3 & 2 & 0 \\ 1 & 5 & 6 & 6 \\ 5 & 3 & 2 & 2 \end{pmatrix}$$

$$\begin{pmatrix} \color{green}{1} & \color{red}{5} & \color{red}{6} & \color{red}{3} \\ 4 & 3 & 2 & 0 \\ 4 & 5 & 6 & 6 \\ 4 & 3 & 2 & 2 \end{pmatrix} \rightarrow$$

Example

$$\begin{pmatrix} 5 & 0 & 6 & 3 \\ 5 & 5 & 2 & 0 \\ 1 & 4 & 6 & 6 \\ 5 & 5 & 2 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 5 & 5 & 6 & 3 \\ 5 & 3 & 2 & 0 \\ 1 & 5 & 6 & 6 \\ 5 & 3 & 2 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 5 & 5 & 6 & 3 \\ 5 & 3 & 2 & 0 \\ 1 & 5 & 6 & 6 \\ 5 & 3 & 2 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 5 & 6 & 3 \\ 4 & 3 & 2 & 0 \\ 4 & 5 & 6 & 6 \\ 4 & 3 & 2 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 4 & 4 & 6 & 2 \\ 4 & 6 & 3 & 1 \\ 4 & 4 & 6 & 4 \end{pmatrix} \rightarrow \left(\begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline 0 & 4 & 6 & 2 \\ 0 & 6 & 3 & 1 \\ 0 & 4 & 6 & 4 \end{array}\right)$$

An efficient algorithm to solve the word problem returning SLPs has been designed:

- SL *Straight-line programs with memory and matrix Bruhat decomposition* by Alice C. Niemeyer, Tomasz Popiel and Cheryl E. Praeger, 2017
- Sp Bachelors thesis, R., 2019
- SU Bachelors thesis, R., 2019
- SO Masters thesis, R. , 2020 (except characteristic 2)

Example in non-natural representation

$$SL(4,7) \cong \langle \begin{pmatrix} 6 & 2 & 3 & 5 & 5 & 1 \\ 5 & 3 & 2 & 3 & 6 & 1 \\ 0 & 0 & 5 & 1 & 3 & 0 \\ 6 & 1 & 0 & 4 & 4 & 4 \\ 2 & 5 & 1 & 3 & 6 & 6 \\ 5 & 2 & 5 & 2 & 3 & 2 \end{pmatrix}, \begin{pmatrix} 5 & 5 & 2 & 0 & 1 & 3 \\ 5 & 2 & 3 & 3 & 1 & 4 \\ 2 & 4 & 2 & 4 & 4 & 5 \\ 5 & 2 & 1 & 6 & 4 & 5 \\ 6 & 3 & 4 & 2 & 3 & 6 \\ 1 & 1 & 5 & 3 & 5 & 4 \end{pmatrix} \rangle =: \hat{G} \leq SL(6,7)$$

Example in non-natural representation

$$\begin{pmatrix} 5 & 0 & 6 & 3 \\ 5 & 5 & 2 & 0 \\ 1 & 4 & 6 & 6 \\ 5 & 5 & 2 & 2 \end{pmatrix} \leftrightarrow \begin{pmatrix} 2 & 3 & 5 & 1 & 0 & 5 \\ 1 & 1 & 2 & 2 & 4 & 6 \\ 4 & 4 & 5 & 2 & 2 & 6 \\ 0 & 4 & 0 & 4 & 4 & 4 \\ 0 & 3 & 2 & 0 & 1 & 2 \\ 5 & 1 & 3 & 5 & 3 & 0 \end{pmatrix} \in \hat{G}$$

$$E_{1,2}(1) \leftrightarrow \begin{pmatrix} 2 & 6 & 5 & 0 & 5 & 3 \\ 2 & 6 & 3 & 0 & 3 & 6 \\ 3 & 4 & 2 & 0 & 1 & 2 \\ 2 & 5 & 3 & 1 & 3 & 6 \\ 2 & 5 & 3 & 0 & 4 & 6 \\ 6 & 1 & 2 & 0 & 2 & 5 \end{pmatrix}$$

Example in non-natural representation

$$E_{1,2}(1) \leftrightarrow \begin{pmatrix} 2 & 6 & 5 & 0 & 5 & 3 \\ 2 & 6 & 3 & 0 & 3 & 6 \\ 3 & 4 & 2 & 0 & 1 & 2 \\ 2 & 5 & 3 & 1 & 3 & 6 \\ 2 & 5 & 3 & 0 & 4 & 6 \\ 6 & 1 & 2 & 0 & 2 & 5 \end{pmatrix} \qquad E_{5,2}(3) \leftrightarrow \begin{pmatrix} 4 & 4 & 1 & 0 & 1 & 2 \\ 6 & 2 & 2 & 0 & 2 & 4 \\ 1 & 6 & 6 & 0 & 5 & 3 \\ 3 & 4 & 1 & 1 & 1 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 4 & 3 & 6 & 0 & 6 & 6 \end{pmatrix}$$

Example in non-natural representation

$$\begin{pmatrix} 5 & \color{green}{5} & 6 & 3 \\ 5 & 3 & 2 & 0 \\ 1 & 5 & 6 & 6 \\ 5 & 3 & 2 & 2 \end{pmatrix} \leftrightarrow \begin{pmatrix} 1 & 4 & 0 & 1 & 2 & 2 \\ 2 & 0 & 0 & 2 & 2 & 2 \\ 5 & 3 & 3 & 2 & 0 & 2 \\ 6 & 5 & 2 & 4 & 6 & 1 \\ 5 & 5 & 6 & 0 & 5 & 3 \\ 2 & 4 & 2 & 5 & 2 & 5 \end{pmatrix} \in \hat{G}$$

Example in non-natural representation

$$\overbrace{\begin{pmatrix} 2 & 3 & 5 & 1 & 0 & 5 \\ 1 & 1 & 2 & 2 & 4 & 6 \\ 4 & 4 & 5 & 2 & 2 & 6 \\ 0 & 4 & 0 & 4 & 4 & 4 \\ 0 & 3 & 2 & 0 & 1 & 2 \\ 5 & 1 & 3 & 5 & 3 & 0 \end{pmatrix}}^{?} \to \begin{pmatrix} 1 & 4 & 0 & 1 & 2 & 2 \\ 2 & 0 & 0 & 2 & 2 & 2 \\ 5 & 3 & 3 & 2 & 0 & 2 \\ 6 & 5 & 2 & 4 & 6 & 1 \\ 5 & 5 & 6 & 0 & 5 & 3 \\ 2 & 4 & 2 & 5 & 2 & 5 \end{pmatrix} \in \hat{G}$$

Constructive Recognition

Let $G = \langle X \rangle \leq \mathrm{GL}(d, q)$ be a a classical group.

Let $G = \langle X \rangle \leq \mathrm{GL}(d, q)$ be a a classical group.
The goal of constructive recognition is to construct an isomorphism

$$\varphi\colon G \to \hat{G}$$

where $\hat{G}$ is the standard copy of G such that $\varphi(g)$ and $\varphi^{-1}(\hat{g})$ can be computed efficiently for each $g \in G$ and $\hat{g} \in \hat{G}$.

Let $G = \langle X \rangle \leq \mathrm{GL}(d, q)$ be a a classical group.
The goal of constructive recognition is to construct an isomorphism

$$\varphi\colon G \to \hat{G}$$

where $\hat{G}$ is the standard copy of G such that $\varphi(g)$ and $\varphi^{-1}(\hat{g})$ can be computed efficiently for each $g \in G$ and $\hat{g} \in \hat{G}$.

- Define a set of *standard generators* $\hat{S}$ for each classical group in the standard copy.

Goal of Constructive Recognition

Let $G = \langle X \rangle \leq \mathrm{GL}(d, q)$ be a a classical group.
The goal of constructive recognition is to construct an isomorphism

$$\varphi\colon G \to \hat{G}$$

where $\hat{G}$ is the standard copy of G such that $\varphi(g)$ and $\varphi^{-1}(\hat{g})$ can be computed efficiently for each $g \in G$ and $\hat{g} \in \hat{G}$.

- Define a set of *standard generators* $\hat{S}$ for each classical group in the standard copy.
- Express $S = \varphi(\hat{S})^{-1}$ as words in terms of X.

Let $G = \langle X \rangle \leq \mathrm{GL}(d, q)$ be a a classical group.
The goal of constructive recognition is to construct an isomorphism

$$\varphi\colon G \to \hat{G}$$

where $\hat{G}$ is the standard copy of G such that $\varphi(g)$ and $\varphi^{-1}(\hat{g})$ can be computed efficiently for each $g \in G$ and $\hat{g} \in \hat{G}$.

- Define a set of *standard generators* $\hat{S}$ for each classical group in the standard copy.
- Express $S = \varphi(\hat{S})^{-1}$ as words in terms of X.
- Express g in terms of S and $\hat{g}$ in terms of $\hat{S}$ as words.

One idea to express the set S in terms of X consists of three steps:

One idea to express the set S in terms of X consists of three steps:

1) GoingDown Algorithm: Reduce the problem to a smaller classical group until a base case.

One idea to express the set S in terms of X consists of three steps:

1) GoingDown Algorithm: Reduce the problem to a smaller classical group until a base case.
2) Base Case Algorithm: Use a constructive recognition algorithm for the base case.

One idea to express the set S in terms of X consists of three steps:

1) GoingDown Algorithm: Reduce the problem to a smaller classical group until a base case.
2) Base Case Algorithm: Use a constructive recognition algorithm for the base case.
3) GoingUp Algorithm: Use the knowledge of standard generators for a smaller classical group to construct standard generators for a larger one.

One idea to express the set S in terms of X consists of three steps:

1) GoingDown Algorithm: Reduce the problem to a smaller SL until a $\mathrm{SL}(2, q)$ is found.
2) Base Case Algorithm: Constructive recognition algorithm for $\mathrm{SL}(2, q)$ by Conder, Leedham-Green and O'Brien.
3) GoingUp Algorithm: Use the knowledge of standard generators for a smaller SL to construct standard generators for a larger SL.

One idea to express the set S in terms of X consists of three steps:

1) GoingDown Algorithm: Reduce the problem to a smaller SL until a $\mathrm{SL}(2, q)$ is found.
2) Base Case Algorithm: Constructive recognition algorithm for $\mathrm{SL}(2, q)$ by Conder, Leedham-Green and O'Brien.
3) GoingUp Algorithm: Use the knowledge of standard generators for a smaller SL to construct standard generators for a larger SL.

GoingDown Algorithm

Let $G \leq \mathrm{GL}(d,q)$ with $G \cong \mathrm{SL}(d,q)$. We search for certain elements in the special linear groups called stingray elements to compute a chain

$$\mathrm{SL}(4,q) \cong U \leq U_k \cong \mathrm{SL}(d_k,q) \leq \ldots \leq U_1 \cong \mathrm{SL}(d_1,q) \leq G$$

where $d_i \leq 2 \cdot \log_2(d_{i-1})$.

Stingray element

Let $G \leq \mathrm{GL}(d, q)$. An element $s \in G$ is called *stingray element*, if s acts irreducibly on a subspace of dimension m, that is it does not leave invariant a subspace of the m-dimensional space, with m small relative to d, and fixes a complementary subspace pointwise.

Example of a Stingray Element

$$\begin{pmatrix} 4 & 1 & 3 & 3 & 2 & 1 & 2 & 4 & 0 & 4 \\ 3 & 3 & 4 & 4 & 3 & 4 & 1 & 0 & 3 & 2 \\ 2 & 1 & 0 & 4 & 3 & 1 & 1 & 2 & 0 & 3 \\ 3 & 1 & 3 & 4 & 4 & 0 & 2 & 0 & 1 & 3 \\ 3 & 4 & 1 & 1 & 0 & 3 & 4 & 4 & 1 & 1 \\ 1 & 0 & 4 & 4 & 2 & 2 & 1 & 1 & 4 & 2 \\ 2 & 4 & 2 & 2 & 0 & 3 & 4 & 2 & 1 & 0 \\ 0 & 2 & 2 & 2 & 4 & 0 & 3 & 4 & 2 & 0 \\ 3 & 4 & 1 & 1 & 4 & 3 & 4 & 4 & 2 & 1 \\ 0 & 2 & 2 & 2 & 2 & 1 & 3 & 2 & 1 & 2 \end{pmatrix} \in \mathrm{SL}(10, 5)$$

Example of a Stingray Element

$$\left(\begin{array}{cccc|cccccc} 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right) \in \mathrm{SL}(10,5)$$

$$\left(\begin{array}{cccc|cccccc}
0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right) \in \mathrm{SL}(10, 5)$$

Stingray pair

$$s_1 = \begin{pmatrix} \boxed{\quad * \quad} \end{pmatrix} \in \mathrm{GL}(d, q) \qquad s_2 = \begin{pmatrix} \boxed{\quad * \quad} \end{pmatrix} \in \mathrm{GL}(d, q)$$

$$s_1 = \begin{pmatrix} \boxed{\quad * \quad} \end{pmatrix} \in \mathrm{GL}(d, q)$$

↓ Change of basis

$$\left(\begin{array}{ccc|ccc} \boxed{*} & & & & & \\ & 1 & & & 0 & \\ & & 1 & & & \\ \hline & & & 1 & \dots & 0 \\ & 0 & & \vdots & \ddots & \vdots \\ & & & 0 & \dots & 1 \end{array}\right)$$

$$s_2 = \begin{pmatrix} \boxed{\quad * \quad} \end{pmatrix} \in \mathrm{GL}(d, q)$$

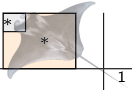
↓ Change of basis

$$\left(\begin{array}{ccc|ccc} \boxed{*} & & & & & \\ & 1 & & & 0 & \\ & & 1 & & & \\ \hline & & & 1 & \dots & 0 \\ & 0 & & \vdots & \ddots & \vdots \\ & & & 0 & \dots & 1 \end{array}\right)$$

Stingray pair

$$s_1 = \left(\begin{array}{|c|} \hline * \\ \hline \end{array} \right) \in \mathrm{GL}(d, q)$$

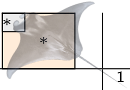
$\downarrow$ Change of basis



$$\left(\begin{array}{c|ccc} \begin{array}{|c|}\hline * \quad \\ \quad * \\ \hline\end{array} & & 0 & \\ \hline & 1 & \cdots & 0 \\ 0 & \vdots & \ddots & \vdots \\ & 0 & \cdots & 1 \end{array} \right)$$

$$s_2 = \left(\begin{array}{|c|} \hline * \\ \hline \end{array} \right) \in \mathrm{GL}(d, q)$$

$\downarrow$ Change of basis

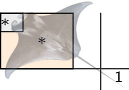


$$\left(\begin{array}{c|ccc} \begin{array}{|c|}\hline * \quad \\ \quad * \\ \hline\end{array} & & 0 & \\ \hline & 1 & \cdots & 0 \\ 0 & \vdots & \ddots & \vdots \\ & 0 & \cdots & 1 \end{array} \right)$$

Stingray pair

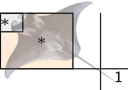
$$s_1 = \left(\begin{array}{|c|}\hline \quad * \quad \\ \hline\end{array}\right) \in \mathrm{GL}(d, q) \qquad s_2 = \left(\begin{array}{|c|}\hline \quad * \quad \\ \hline\end{array}\right) \in \mathrm{GL}(d, q)$$

$\downarrow$ Change of basis



$$\left(\begin{array}{c|c} \begin{array}{|cc|}\hline * & \\ & * \\ \hline\end{array} & 0 \\ \hline 0 & \begin{array}{ccc} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{array} \end{array}\right)$$

$\downarrow$ Change of basis



$$\left(\begin{array}{c|c} \begin{array}{|cc|}\hline * & \\ & * \\ \hline\end{array} & 0 \\ \hline 0 & \begin{array}{ccc} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{array} \end{array}\right)$$

$$\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}$$

$$\left(\begin{array}{c|c} \begin{array}{|c|}\hline \mathrm{SL}(n, q) \\ \hline\end{array} & 0 \\ \hline 0 & \begin{array}{ccc} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{array} \end{array}\right)$$

Stingray candidate

Let $G \leq \mathrm{GL}(d, q)$. An element $s' \in G$ is called *stingray candidate*, if s' acts on a subspace of dimension m with m small relative to d, and fixes a complementary subspace pointwise.

Stingray candidate

Let $G \leq \mathrm{GL}(d, q)$. An element $s' \in G$ is called *stingray candidate*, if s' acts on a subspace of dimension m with m small relative to d, and fixes a complementary subspace pointwise.

Pre-stingray candidate

Let $G \leq \mathrm{GL}(d, q)$. An element $\tilde{s} \in G$ is called *pre-stingray candidate*, if the characteristic polynomial $\chi_{\tilde{s}}(x)$ has an irreducible factor $P(x) \in \mathbb{F}_q[x]$ of degree k over $\mathbb{F}_q$ and no other irreducible factors of degree divisible by k.

Pre-stingray candidate

Theorem

Let $G \leq \mathrm{GL}(d, q)$ and $\tilde{s} \in G$ a pre-stingray candidate. Then $\tilde{s}^{\ell}$ is a stingray candidate where $\ell \in \mathbb{N}$ can easily be computed with information from $\chi_{\tilde{s}}(x)$.

Find Stingray Candidate

Algorithm 1 FindStingrayCandidate

procedure FindStingrayCandidate(H, n)
  while true **do**
    $h :=$ (pseudo)random(H); $\chi_h(x) :=$ characteristic polynomial of h;
    $\{P_i(x)\} :=$ irreducible factors of $\chi_h(x)$;
    if $\exists P_i(x)$, $k := \deg(P_i(x))$ does not divide $\deg(P_j(x)), j \neq i$ **then**
      Compute $\ell \in \mathbb{N}$ according to theorem
      Return h^ℓ;
    end if
  end do

Construct a smaller SL

Let $s_1, s_2 \in G \leq \mathrm{GL}(d, q)$ with $G \cong \mathrm{SL}(d, q)$ and s_1, s_2 stingray elements where s_i acts irreducibly on $W_i \leq \mathbb{F}_q^d$ and $W_1 \cap W_2 = \{0\}$. With some probability

$$\langle s_1, s_2 \rangle \cong \mathrm{SL}(m_1 + m_2, q)$$

where m_i is the dimension of W_i.

Probability

1) What is the probability to find a pre-stingray candidate?

1) What is the probability to find a pre-stingray candidate?
Elements in finite classical groups whose powers have large 1-Eigenspaces by Alice C. Niemeyer and Cheryl E. Praeger, 2018

Probability

1) What is the probability to find a pre-stingray candidate?
Elements in finite classical groups whose powers have large 1-Eigenspaces by Alice C. Niemeyer and Cheryl E. Praeger, 2018
2) What is the probability that two stingray elements generate a special linear group?

1) What is the probability to find a pre-stingray candidate?
Elements in finite classical groups whose powers have large 1-Eigenspaces by Alice C. Niemeyer and Cheryl E. Praeger, 2018
2) What is the probability that two stingray elements generate a special linear group?
The probability of spanning a classical space by two non-degenerate subspaces of complementary dimensions by S.P. Glasby, Alice C. Niemeyer and Cheryl E. Praeger
Random generation of direct sums of finite non-degenerate subspaces by S.P. Glasby, Alice C. Niemeyer and Cheryl E. Praeger

Second item is not completely solved yet.

Thank you for your attention!