Matrix Groups

Max Neunhöffer

University of St Andrews

GAC 2010, Allahabad

Introduction

Let $\mathbb{F}$ be a field. Set

$$\mathrm{GL}_d(\mathbb{F}) := \left\{ M \in \mathbb{F}^{d \times d} \mid M \text{ is invertible} \right\}.$$

This is a group under matrix multiplication.

Introduction

Let $\mathbb{F}$ be a field. Set

$$\mathrm{GL}_d(\mathbb{F}) := \left\{ M \in \mathbb{F}^{d \times d} \mid M \text{ is invertible} \right\}.$$

This is a group under matrix multiplication.

We are here particularly interested in the case that $\mathbb{F} = \mathbb{F}_q$ is the finite field with $q = p^f$ elements.

Introduction

Let $\mathbb{F}$ be a field. Set

$$\mathrm{GL}_d(\mathbb{F}) := \left\{ M \in \mathbb{F}^{d\times d} \mid M \text{ is invertible} \right\}.$$

This is a group under matrix multiplication.

We are here particularly interested in the case that $\mathbb{F} = \mathbb{F}_q$ is the finite field with $q = p^f$ elements.

Definition (Matrix group, projective group)

A matrix group is a subgroup of some $\mathrm{GL}_d(\mathbb{F})$.

Introduction

Let $\mathbb{F}$ be a field. Set

$$\mathrm{GL}_d(\mathbb{F}) := \left\{ M \in \mathbb{F}^{d \times d} \mid M \text{ is invertible} \right\}.$$

This is a group under matrix multiplication.
We are here particularly interested in the case that $\mathbb{F} = \mathbb{F}_q$ is the finite field with $q = p^f$ elements.

Definition (Matrix group, projective group)

A matrix group is a subgroup of some $\mathrm{GL}_d(\mathbb{F})$.

We call two matrices $M, N \in \mathrm{GL}_d(\mathbb{F})$ equivalent, if one is a scalar multiple of the other and denote the equivalence class of M by $[M]$. Then

$$\mathrm{PGL}_d(\mathbb{F}) := \{[M] \mid M \in \mathrm{GL}_d(\mathbb{F})\}$$

is a group with the well-defined multiplication

$$[M] \cdot [N] := [MN].$$

This is called the projective group.

GAP examples

see other window

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1\times d}$,

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1\times d}$,
- V modulo scalars (projective action),

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1 \times d}$,
- V modulo scalars (projective action),
- $\{W \leq V \mid \dim_{\mathbb{F}_q} W = k\}$ for some $1 \leq k < d$.

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1\times d}$,
- V modulo scalars (projective action),
- $\{W \leq V \mid \dim_{\mathbb{F}_q} W = k\}$ for some $1 \leq k < d$.

If $vG \subseteq V$ is an orbit, then we have a group homomorphism

$$\psi : G \to \Sigma_{vG}, g \mapsto (vG \to vG, vh \mapsto vhg).$$

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1\times d}$,
- V modulo scalars (projective action),
- $\{W \leq V \mid \dim_{\mathbb{F}_q} W = k\}$ for some $1 \leq k < d$.

If $vG \subseteq V$ is an orbit, then we have a group homomorphism

$$\psi : G \to \Sigma_{vG}, g \mapsto (vG \to vG, vh \mapsto vhg).$$

Lemma

If vG contains a basis of V, then ψ is injective.

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1 \times d}$,
- V modulo scalars (projective action),
- $\{W \leq V \mid \dim_{\mathbb{F}_q} W = k\}$ for some $1 \leq k < d$.

If $vG \subseteq V$ is an orbit, then we have a group homomorphism

$$\psi : G \to \Sigma_{vG}, g \mapsto (vG \to vG, vh \mapsto vhg).$$

Lemma

If vG contains a basis of V, then ψ is injective.

In this case, we can

- explicitly compute the image of $g \in G$ by acting,

Matrix groups in GAP(currently!)

GAP handles matrix groups via permutation groups:

Let $G \leq \mathrm{GL}_d(\mathbb{F}_q)$. Then G acts linearly on

- $V := \mathbb{F}_q^{1\times d}$,
- V modulo scalars (projective action),
- $\{W \leq V \mid \dim_{\mathbb{F}_q} W = k\}$ for some $1 \leq k < d$.

If $vG \subseteq V$ is an orbit, then we have a group homomorphism

$$\psi : G \to \Sigma_{vG}, g \mapsto (vG \to vG, vh \mapsto vhg).$$

Lemma

If vG contains a basis of V, then ψ is injective.

In this case, we can

- explicitly compute the image of $g \in G$ by acting,
- explicitly compute the preimage of a permutation by reading off the images of the basis vectors in vG.

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix groups act on lots of stuff, just pick an orbit!

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix groups act on lots of stuff, just pick an orbit!

One problem is to find a short orbit!

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix groups act on lots of stuff, just pick an orbit!

One problem is to find a short orbit!

Various heuristics are used, for example try vectors in intersections of eigenspaces of random elements, but in general, this is difficult.

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix groups act on lots of stuff, just pick an orbit!

One problem is to find a short orbit!

Various heuristics are used, for example try vectors in intersections of eigenspaces of random elements, but in general, this is difficult.

One usually uses projective action and action on vectors alternatingly.

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix groups act on lots of stuff, just pick an orbit!

One problem is to find a short orbit!

Various heuristics are used, for example try vectors in intersections of eigenspaces of random elements, but in general, this is difficult.

One usually uses projective action and action on vectors alternatingly.

Magma has lots of algorithms for matrix groups using stabiliser chains.

Matrix Schreier-Sims

In principle one can use the Schreier-Sims procedure to compute a stabiliser chain for matrix groups as well.

Matrix groups act on lots of stuff, just pick an orbit!

One problem is to find a short orbit!

Various heuristics are used, for example try vectors in intersections of eigenspaces of random elements, but in general, this is difficult.

One usually uses projective action and action on vectors alternatingly.

Magma has lots of algorithms for matrix groups using stabiliser chains.

For GAP there is the genss package to compute stabiliser chains but not yet many algorithms to use them.

Problem: very big orbits

A fundamental problem with both approaches is the following:

Problem (Giants)

For $G = \mathrm{GL}_d(\mathbb{F}_q)$, the shortest non-trivial orbit in V is $V \setminus \{0\}$ with $q^d - 1$ elements.

Problem: very big orbits

A fundamental problem with both approaches is the following:

Problem (Giants)

For $G = \mathrm{GL}_d(\mathbb{F}_q)$, the shortest non-trivial orbit in V is $V \setminus \{0\}$ with $q^d - 1$ elements.
In projective action the length is $(q^d - 1)/(q - 1)$ which is only slightly better.

Problem: very big orbits

A fundamental problem with both approaches is the following:

Problem (Giants)

For $G = \mathrm{GL}_d(\mathbb{F}_q)$, the shortest non-trivial orbit in V is $V \setminus \{0\}$ with $q^d - 1$ elements.
In projective action the length is $(q^d - 1)/(q - 1)$ which is only slightly better.

So for the whole $\mathrm{GL}_d(\mathbb{F}_q)$, there are no short orbits!

Problem: very big orbits

A fundamental problem with both approaches is the following:

Problem (Giants)

For $G = \mathrm{GL}_d(\mathbb{F}_q)$, the shortest non-trivial orbit in V is $V \setminus \{0\}$ with $q^d - 1$ elements.
In projective action the length is $(q^d - 1)/(q - 1)$ which is only slightly better.

So for the whole $\mathrm{GL}_d(\mathbb{F}_q)$, there are no short orbits!

Like in the Σ_n case, each $\mathrm{GL}_d(\mathbb{F}_q)$ contains certain large subgroups with this problem. They are called the classical groups in their natural representation. Examples:

- the special linear group $\mathrm{SL}_d(\mathbb{F}_q)$,

Problem: very big orbits

A fundamental problem with both approaches is the following:

Problem (Giants)

For $G = \mathrm{GL}_d(\mathbb{F}_q)$, the shortest non-trivial orbit in V is $V \setminus \{0\}$ with $q^d - 1$ elements.
In projective action the length is $(q^d - 1)/(q - 1)$ which is only slightly better.

So for the whole $\mathrm{GL}_d(\mathbb{F}_q)$, there are no short orbits!

Like in the Σ_n case, each $\mathrm{GL}_d(\mathbb{F}_q)$ contains certain large subgroups with this problem. They are called the classical groups in their natural representation. Examples:

- the special linear group $\mathrm{SL}_d(\mathbb{F}_q)$,
- the symplectic group $\mathrm{Sp}_{2d}(\mathbb{F}_q)$,

Problem: very big orbits

A fundamental problem with both approaches is the following:

Problem (Giants)

For $G = \mathrm{GL}_d(\mathbb{F}_q)$, the shortest non-trivial orbit in V is $V \setminus \{0\}$ with $q^d - 1$ elements.
In projective action the length is $(q^d - 1)/(q - 1)$ which is only slightly better.

So for the whole $\mathrm{GL}_d(\mathbb{F}_q)$, there are no short orbits!

Like in the Σ_n case, each $\mathrm{GL}_d(\mathbb{F}_q)$ contains certain large subgroups with this problem. They are called the classical groups in their natural representation. Examples:

- the special linear group $\mathrm{SL}_d(\mathbb{F}_q)$,
- the symplectic group $\mathrm{Sp}_{2d}(\mathbb{F}_q)$,
- the unitary group $\mathrm{U}_d(\mathbb{F}_{q^2})$.

Matrix groups and group algebras

Definition (Group algebra)

Let G be a finite group and $\mathbb{F}$ a field. Then $\mathbb{F}G$, the group algebra, is

- an $\mathbb{F}$-vector space with basis G, and
- multiplication $(\sum_{g \in G} a_g g) \cdot (\sum_{h \in G} b_h h) := \sum_{g,h \in G} a_g b_h \cdot gh$.

$\mathbb{F}G$ is an $\mathbb{F}$-algebra.

Matrix groups and group algebras

Definition (Group algebra)

Let G be a finite group and $\mathbb{F}$ a field. Then $\mathbb{F}G$, the group algebra, is

- an $\mathbb{F}$-vector space with basis G, and
- multiplication $(\sum_{g \in G} a_g g) \cdot (\sum_{h \in G} b_h h) := \sum_{g,h \in G} a_g b_h \cdot gh$.

$\mathbb{F}G$ is an $\mathbb{F}$-algebra.

Proposition

A group homomorphism $G \to \mathrm{GL}_d(\mathbb{F})$ "amounts to the same" as an $\mathbb{F}G$-module of dimension d.

Matrix groups and group algebras

Definition (Group algebra)

Let G be a finite group and $\mathbb{F}$ a field. Then $\mathbb{F}G$, the group algebra, is

- an $\mathbb{F}$-vector space with basis G, and
- multiplication $(\sum_{g \in G} a_g g) \cdot (\sum_{h \in G} b_h h) := \sum_{g,h \in G} a_g b_h \cdot gh$.

$\mathbb{F}G$ is an $\mathbb{F}$-algebra.

Proposition

A group homomorphism $G \to \mathrm{GL}_d(\mathbb{F})$ "amounts to the same" as an $\mathbb{F}G$-module of dimension d.

Idea of proof: "$\to$": Extend the action of G via $\mathrm{GL}_d(\mathbb{F})$ on $\mathbb{F}^{1 \times d}$ linearly to $\mathbb{F}G$. "$\leftarrow$": Restrict action to basis. ∎

Matrix groups and group algebras

Definition (Group algebra)

Let G be a finite group and $\mathbb{F}$ a field. Then $\mathbb{F}G$, the group algebra, is

- an $\mathbb{F}$-vector space with basis G, and
- multiplication $(\sum_{g\in G} a_g g)\cdot(\sum_{h\in G} b_h h) := \sum_{g,h\in G} a_g b_h \cdot gh$.

$\mathbb{F}G$ is an $\mathbb{F}$-algebra.

Proposition

A group homomorphism $G \to \mathrm{GL}_d(\mathbb{F})$ "amounts to the same" as an $\mathbb{F}G$-module of dimension d.

Idea of proof: "$\to$": Extend the action of G via $\mathrm{GL}_d(\mathbb{F})$ on $\mathbb{F}^{1\times d}$ linearly to $\mathbb{F}G$. "$\leftarrow$": Restrict action to basis. ∎

$\Longrightarrow$ Can use the MeatAxe for matrix groups $G \leq \mathrm{GL}_d(\mathbb{F}_q)$.

Matrix groups and group algebras

Definition (Group algebra)

Let G be a finite group and $\mathbb{F}$ a field. Then $\mathbb{F}G$, the group algebra, is

- an $\mathbb{F}$-vector space with basis G, and
- multiplication $(\sum_{g \in G} a_g g) \cdot (\sum_{h \in G} b_h h) := \sum_{g,h \in G} a_g b_h \cdot gh$.

$\mathbb{F}G$ is an $\mathbb{F}$-algebra.

Proposition

A group homomorphism $G \to \mathrm{GL}_d(\mathbb{F})$ "amounts to the same" as an $\mathbb{F}G$-module of dimension d.

Idea of proof: "$\to$": Extend the action of G via $\mathrm{GL}_d(\mathbb{F})$ on $\mathbb{F}^{1 \times d}$ linearly to $\mathbb{F}G$. "$\leftarrow$": Restrict action to basis. ∎

$\Longrightarrow$ Can use the MeatAxe for matrix groups $G \leq \mathrm{GL}_d(\mathbb{F}_q)$.

Distinguish between $\langle G \rangle_{\mathrm{Alg}} \leq \mathbb{F}_q^{d \times d}$ and $\langle G \rangle_{\mathrm{Alg}} \leq \mathbb{F}_q G$!

Straight line programs

Definition (Straight line program)

Informally: a program with no branches.

Straight line programs

Definition (Straight line program)

Informally: a program with no branches.

More formally:

- The input is a finite list of group elements.

Straight line programs

Definition (Straight line program)

Informally: a program with no branches.

More formally:

- The input is a finite list of group elements.
- The program consists of a finite list of steps.

Straight line programs

Definition (Straight line program)

Informally: a program with no branches.

More formally:

- The input is a finite list of group elements.
- The program consists of a finite list of steps.
- Each step only computes a product of powers of previously acquired group elements.

Straight line programs

Definition (Straight line program)

Informally: a program with no branches.

More formally:

- The input is a finite list of group elements.
- The program consists of a finite list of steps.
- Each step only computes a product of powers of previously acquired group elements.
- The output is a finite list of the results.

Straight line programs

Definition (Straight line program)

Informally: a program with no branches.

More formally:

- The input is a finite list of group elements.
- The program consists of a finite list of steps.
- Each step only computes a product of powers of previously acquired group elements.
- The output is a finite list of the results.

An example: (computes a commutator)

```
# input:
r:= [ g1, g2 ];
# program:
r[3]:= r[1]↑-1;
r[4]:= r[2]↑-1;
r[5]:= r[1]*r[2]*r[3]*r[4];
# return value:
r[5]
```

Constructive recognition

Problem

Let $\mathbb{F}_q$ be the field with q elements und

$$M_1, \ldots, M_k \in \mathrm{GL}_n(\mathbb{F}_q).$$

Find for $G := \langle M_1, \ldots, M_k \rangle$:

- The group order $|G|$ and
- an algorithm that, given $M \in \mathrm{GL}_n(\mathbb{F}_q)$,
  - decides, whether or not $M \in G$, and,
  - if so, expresses M as an SLP in the M_i.

Constructive recognition

Problem

Let $\mathbb{F}_q$ be the field with q elements und

$$M_1, \ldots, M_k \in \mathrm{GL}_n(\mathbb{F}_q).$$

Find for $G := \langle M_1, \ldots, M_k \rangle$:

- The group order $|G|$ and
- an algorithm that, given $M \in \mathrm{GL}_n(\mathbb{F}_q)$,
  - decides, whether or not $M \in G$, and,
  - if so, expresses M as an SLP in the M_i.
- The runtime should be bounded from above by a polynomial in n, k and $\log q$.

Constructive recognition

Problem

Let $\mathbb{F}_q$ be the field with q elements und

$$M_1, \ldots, M_k \in \mathrm{GL}_n(\mathbb{F}_q).$$

Find for $G := \langle M_1, \ldots, M_k \rangle$:

- The group order $|G|$ and
- an algorithm that, given $M \in \mathrm{GL}_n(\mathbb{F}_q)$,
  - decides, whether or not $M \in G$, and,
  - if so, expresses M as an SLP in the M_i.
- The runtime should be bounded from above by a polynomial in n, k and $\log q$.
- A Monte Carlo Algorithmus is enough.

Constructive recognition

Problem

Let $\mathbb{F}_q$ be the field with q elements und

$$M_1, \ldots, M_k \in \mathrm{GL}_n(\mathbb{F}_q).$$

Find for $G := \langle M_1, \ldots, M_k \rangle$:

- The group order $|G|$ and
- an algorithm that, given $M \in \mathrm{GL}_n(\mathbb{F}_q)$,
  - decides, whether or not $M \in G$, and,
  - if so, expresses M as an SLP in the M_i.
- The runtime should be bounded from above by a polynomial in n, k and $\log q$.
- A Monte Carlo Algorithmus is enough. (Verification!)

Constructive recognition

Problem

Let $\mathbb{F}_q$ be the field with q elements und

$$M_1, \ldots, M_k \in \mathrm{GL}_n(\mathbb{F}_q).$$

Find for $G := \langle M_1, \ldots, M_k \rangle$:

- The group order $|G|$ and
- an algorithm that, given $M \in \mathrm{GL}_n(\mathbb{F}_q)$,
  - decides, whether or not $M \in G$, and,
  - if so, expresses M as an SLP in the M_i.
- The runtime should be bounded from above by a polynomial in n, k and $\log q$.
- A Monte Carlo Algorithmus is enough. (Verification!)

If this problem is solved, we call

$\langle M_1, \ldots, M_k \rangle$ recognised constructively.

Troubles

The discrete logarithm problem

If $M_1 = [z] \in \mathbb{F}_q^{1\times 1}$ with z a primitive root of $\mathbb{F}_q$. Then:

$$\text{Given } 0 \neq [x] \in \mathbb{F}_q^{1\times 1}, \text{ find } i \in \mathbb{N} \text{ such that } [x] = [z]^i.$$

Troubles

The discrete logarithm problem

If $M_1 = [z] \in \mathbb{F}_q^{1\times 1}$ with z a primitive root of $\mathbb{F}_q$. Then:

Given $0 \neq [x] \in \mathbb{F}_q^{1\times 1}$, find $i \in \mathbb{N}$ such that $[x] = [z]^i$.

There is no solution in polynomial time in $\log q$ known!

Troubles

The discrete logarithm problem

If $M_1 = [z] \in \mathbb{F}_q^{1\times 1}$ with z a primitive root of $\mathbb{F}_q$. Then:

Given $0 \neq [x] \in \mathbb{F}_q^{1\times 1}$, find $i \in \mathbb{N}$ such that $[x] = [z]^i$.

There is no solution in polynomial time in $\log q$ known!

Integer factorisation

Some methods need a factorisation of $q^i - 1$ for an $i \leq n$.

Troubles

The discrete logarithm problem

If $M_1 = [z] \in \mathbb{F}_q^{1\times 1}$ with z a primitive root of $\mathbb{F}_q$. Then:

$$\text{Given } 0 \neq [x] \in \mathbb{F}_q^{1\times 1}, \text{ find } i \in \mathbb{N} \text{ such that } [x] = [z]^i.$$

There is no solution in polynomial time in $\log q$ known!

Integer factorisation

Some methods need a factorisation of $q^i - 1$ for an $i \leq n$.

There is no solution in polynomial time in $\log q$ known!

Troubles

The discrete logarithm problem

If $M_1 = [z] \in \mathbb{F}_q^{1\times 1}$ with z a primitive root of $\mathbb{F}_q$. Then:

$$\text{Given } 0 \neq [x] \in \mathbb{F}_q^{1\times 1}, \text{ find } i \in \mathbb{N} \text{ such that } [x] = [z]^i.$$

There is no solution in polynomial time in $\log q$ known!

Integer factorisation

Some methods need a factorisation of $q^i - 1$ for an $i \leq n$.

There is no solution in polynomial time in $\log q$ known!

In practice q is small $\Rightarrow$ no problem.

We ignore both!

The End

Bibliography

Derek F. Holt, Bettina Eick, and Eamonn A. O'Brien. *Handbook of computational group theory*. Discrete Mathematics and its Applications (Boca Raton). Chapman & Hall/CRC, Boca Raton, FL, 2005.