Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

he problei

Randomised algorithm
Constructive recognition

Reduction

Homomorphiem

Computing the kernel

Recursion: compositio

Example: invariant

oacopado

rinding reduction

Solution for leaves

Classifications

Verification

Matrix group recognition

Max Neunhöffer



University of St Andrews

St Andrews, 26.11.2007

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

he problei

Randomised algorithm
Constructive recognition

Reduction

Hamamarahiama

Computing the kernel Recursion: composition

Example: invariant

Finding reduction

Solution for leaves

Siassifications Recognition of the group

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel
Recursion: compositior
trees
Example: invariant

subspace Finding reducti

Finding reduction

Solution for leaves

Recognition of the group

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Finding reduct

Solution for leaves

Varification

Matrix groups ...

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reducti

Solution for leave:

Recognition of the group

Matrix groups ...

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Finding reduction

Solution for leaves Classifications

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

What do we want to determine about *G*?

The group order |G|

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees

subspace Finding reducti

Solution for leave

Recognition of the gr

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given: $M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

- The group order |G|
- Membership test: Is $M \in GL_n(\mathbb{F}_q)$ in G?

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees

subspace

Finding reduction

Solution for leave

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

- The group order |G|
- Membership test: Is $M \in GL_n(\mathbb{F}_q)$ in G?
- Homomorphisms $\varphi : G \rightarrow H$?

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Solution for leave

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

- The group order |G|
- Membership test: Is $M \in GL_n(\mathbb{F}_q)$ in G?
- Homomorphisms $\varphi : G \rightarrow H$?
- Kernels of homomorphisms? Is G simple?

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reduction

Solution for leave

Recognition of the group

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given: $M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

- The group order |G|
- Membership test: Is $M \in GL_n(\mathbb{F}_q)$ in G?
- Homomorphisms $\varphi : G \rightarrow H$?
- Kernels of homomorphisms? Is G simple?
- Comparison with known groups

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reducti

Solution for leaves Classifications

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given: $M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

- The group order |G|
- Membership test: Is $M \in GL_n(\mathbb{F}_q)$ in G?
- Homomorphisms $\varphi : G \rightarrow H$?
- Kernels of homomorphisms? Is *G* simple?
- Comparison with known groups
- (Maximal) subgroups?

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reduction

Solution for leave

Verification

Matrix groups . . .

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

$$\operatorname{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given: $M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

It is finite, we have $|GL_n(\mathbb{F}_q)| = q^{n(n-1)/2} \prod_{i=1}^n (q^i - 1)$

- The group order |G|
- Membership test: Is $M \in GL_n(\mathbb{F}_q)$ in G?
- Homomorphisms $\varphi : G \rightarrow H$?
- Kernels of homomorphisms? Is G simple?
- Comparison with known groups
- (Maximal) subgroups?
- ...

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Solution for leave

Recognition of the groups

Verification

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

$$\mathrm{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given: $M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$

Then the M_i generate a group $G \leq \operatorname{GL}_n(\mathbb{F}_q)$.

Introduction

Matrix groups

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: compositior
trees
Example: invariant

Solution for leave

V----

Permutation groups and matrix groups

Let $n \in \mathbb{N}$ and S_n be the symmetric group:

$$\mathcal{S}_n = \{\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\} \mid \pi \text{ bijective}\}.$$

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

$$\mathrm{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Solution for leave

Verification

Permutation groups and matrix groups

Let $n \in \mathbb{N}$ and S_n be the symmetric group:

$$S_n = \{\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\} \mid \pi \text{ bijective}\}.$$

Given:
$$\pi_1, \ldots, \pi_k \in S_n$$

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

$$\mathrm{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Solution for leave

Verification

Permutation groups and matrix groups

Let $n \in \mathbb{N}$ and S_n be the symmetric group:

$$\mathcal{S}_n = \{\pi: \{1,\dots,n\} \rightarrow \{1,\dots,n\} \mid \pi \text{ bijective}\}.$$

Given:
$$\pi_1, \ldots, \pi_k \in S_n$$

Then the π_i generate a group $G \leq S_n$.

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

$$\mathrm{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given:
$$M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leaves

Verification

Permutation groups and matrix groups

Let $n \in \mathbb{N}$ and S_n be the symmetric group:

$$S_n = \{\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\} \mid \pi \text{ bijective}\}.$$

Given: $\pi_1, \ldots, \pi_k \in S_n$

Then the π_i generate a group $G \leq S_n$.

It is finite, we have $|S_n| = n!$

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

$$\mathrm{GL}_n(\mathbb{F}_q) := \{ M \in \mathbb{F}_q^{n \times n} \mid M \text{ invertible} \}$$

Given: $M_1, \ldots, M_k \in \operatorname{GL}_n(\mathbb{F}_q)$

Then the M_i generate a group $G \leq GL_n(\mathbb{F}_q)$.

Introduction

Matrix groups

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms Computing the kernel Recursion: composition trees

Example: invar subspace

Finding reduct

Solution for leave

Verification

Permutation groups

Let $n \in \mathbb{N}$ and S_n be the symmetric group:

$$S_n = \{\pi : \{1,\ldots,n\} \rightarrow \{1,\ldots,n\} \mid \pi \text{ bijective}\}.$$

Given: $\pi_1, \ldots, \pi_k \in S_n$

Then the π_i generate a group $G \leq S_n$.

It is finite, we have $|S_n| = n!$.

We can determine about G algorithmically (e.g.):

- The group order |G|
- Membership test: Is $M \in S_n$ in G?
- Homomorphisms $\varphi : G \rightarrow H$?
- Kernels of homomorphisms? Is *G* simple?
- Comparison with known groups
- (Maximal) subgroups?
-

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognitio
Troubles

Reduction

Computing the kernel
Recursion: compositio
trees

Example: invariant subspace
Finding reductions

Solution for leave

Classifications
Recognition of the groups

Verification

Constructive recognition — first formulation

Problem

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

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

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

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

Introduction

Constructive recognition

The problem

Reduction

Constructive recognition — first formulation

Problem

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

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

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

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

If this problem is solved, we call

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

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel Recursion: composition trees Example: invariant subspace

Solution for leave
Classifications

Varification

Complexity of algorithms

To measure the efficiency of an algorithm, we consider a class \mathcal{P} of problems, that the algorithm can solve.

We assign to each $P \in \mathcal{P}$ its size g(P),

and prove an upper bound for the runtime L(P) of the algorithm for P:

$$L(P) \leq f(g(P))$$

for some function f.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leave

Verification

Complexity of algorithms

To measure the efficiency of an algorithm, we consider a class \mathcal{P} of problems, that the algorithm can solve.

We assign to each $P \in \mathcal{P}$ its size g(P),

and prove an upper bound for the runtime L(P) of the algorithm for P:

$$L(P) \leq f(g(P))$$

for some function *f*.

The growth rate of *f* measures the complexity.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leave:
Classifications
Recognition of the groups

Verification

Complexity of algorithms

To measure the efficiency of an algorithm, we consider a class \mathcal{P} of problems, that the algorithm can solve.

We assign to each $P \in \mathcal{P}$ its size g(P),

and prove an upper bound for the runtime L(P) of the algorithm for P:

$$L(P) \leq f(g(P))$$

for some function *f*.

The growth rate of *f* measures the complexity.

Example (Constructive matrix group recognition)

- Problem given by $M_1, \ldots, M_k \in \mathrm{GL}_n(\mathbb{F}_q)$.
- Size determined by n, k and log q.
- Runtime should be \leq a polynomial in n, k and $\log q$.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The probler

Randomised algorithms

Constructive recognit

Reduction

Homomorphieme

Homomorphism

Pacureian: compact

trees

subspace

Finding reduction

Solution for leaves

Diassifications

Verification

Randomised algorithms

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for le

Classifications
Recognition of the groups

Verification

Randomised algorithms

Definition (Monte Carlo algorithms)

A Monte Carlo algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it returns a wrong result is at most ϵ .

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reducti

- . . .

Solution for leaves Classifications

Recognition of the

Randomised algorithms

Definition (Monte Carlo algorithms)

A Monte Carlo algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it returns a wrong result is at most ϵ .

Definition (Las Vegas algorithm)

A Las Vegas algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it fails is at most ϵ .

Introduction

The problem

Randomised algorithms

Reduction

Randomised algorithms

Definition (Monte Carlo algorithms)

A Monte Carlo algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it returns a wrong result is at most ϵ .

Definition (Las Vegas algorithm)

A Las Vegas algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it fails is at most ϵ .

Example: Comp. of |G| = 4089470473293004800 for permutation group $G = \langle \pi_1, \pi_2 \rangle$ (n = 137632):

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

subspace Finding reducti

Finding reduction

Classifications

Verification

Randomised algorithms

Definition (Monte Carlo algorithms)

A Monte Carlo algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it returns a wrong result is at most ϵ .

Definition (Las Vegas algorithm)

A Las Vegas algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it fails is at most ϵ .

Example: Comp. of |G| = 4089470473293004800 for permutation group $G = \langle \pi_1, \pi_2 \rangle$ (n = 137632): deterministic alg.: 112s

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

subspace Finding reduction

Solution for lea

Recognition of the group

Verification

Randomised algorithms

Definition (Monte Carlo algorithms)

A Monte Carlo algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it returns a wrong result is at most ϵ .

Definition (Las Vegas algorithm)

A Las Vegas algorithm with error probability ϵ is an algorithm, that is guaranteed to terminate after a finite time, such that the probability that it fails is at most ϵ .

Example: Comp. of |G|=4089470473293004800 for permutation group $G=\langle \pi_1,\pi_2\rangle$ (n=137632): deterministic alg.: 112s Monte Carlo $\epsilon=1\%$: 6s

Saving: 95% of runtime

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel Recursion: composition trees

Example: invariant subspace

Finding reduction

Solution for leave

Recognition of the group

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 $M \in GL_n(\mathbb{F}_q)$,
 - decides, whether or not $M \in G$, and,
 - if so, expresses M as word in the M_i .

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invaria subspace Finding reduction

Solution for leave

Classifications
Recognition of the groups

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 word in the M_i .
- The runtime should be bounded from above by a polynomial in n, k and log q.

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invariant subspace

Finding reducti

Classifications Recognition of the groups

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 $G := \langle M_1, \dots, M_k \rangle$:

- The group order |G| and
- an algorithm that, given $M \in GL_n(\mathbb{F}_q)$,
 - decides, whether or not $M \in G$, and,
 - if so, expresses *M* as word 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.

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithms
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reducti

Classifications

Becognition of the groups

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 $M \in GL_n(\mathbb{F}_q)$,
 - decides, whether or not $M \in G$, and.
 - if so, expresses M as word 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!)

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithms
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reducti

Solution for leave:
Classifications
Recognition of the groups

Varification

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 $M \in GL_n(\mathbb{F}_q)$,
 - decides, whether or not $M \in G$, and,
 - if so, expresses M as word 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.

Max Neunhöffer

Introduction

Troubles

Reduction

Troubles

The discrete logarithm problem

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

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

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm:
Constructive recognitio
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invaria subspace Finding reduction

Finding reduction

Solution for leave

Recognition of the group

Verification

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!

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognitio
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reduction

Solution for le

Classifications

Verification

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 $i \le n$.

Max Neunhöffer

Introduction

The problem

Troubles

Reduction

Troubles

The discrete logarithm problem

If $M_1 = [z] \in \mathbb{F}_a^{1 \times 1}$ with z a primitive root of \mathbb{F}_a . 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 $i \le n$.

There is no solution in polynomial time in log *q* known!

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognitic
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Finding reduction

Solution for leave

Varification

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 $i \le n$.

There is no solution in polynomial time in log *q* known!

In practice q is small \Rightarrow no problem. We ignore both!

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Randomised algorithm
Constructive recognition

Reduction

Homomorphisms

Computing the kernel Recursion: composition trees

subspace

Finding reduction

Solution for leaves

Recognition of the group

Verification

What is a reduction?

Let
$$G := \langle M_1, \dots, M_k \rangle \leq \operatorname{GL}_n(\mathbb{F}_q)$$
.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The probler

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel Recursion: compositio trees

Example: invariant subspace

Finding reducti

Solution for leaves

Recognition of the group

Verification

What is a reduction?

Let
$$G := \langle M_1, \dots, M_k \rangle \leq \operatorname{GL}_n(\mathbb{F}_q)$$
.

A reduction is a group homomorphism

$$\varphi: G \to H$$
 $M_i \mapsto P_i$ for all i

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel Recursion: composition trees

Example: invariant subspace

Finding reducti

Solution for leave

Recognition of the group

Verification

What is a reduction?

Let
$$G := \langle M_1, \dots, M_k \rangle \leq \operatorname{GL}_n(\mathbb{F}_q)$$
.

A reduction is a group homomorphism

$$\varphi: G \to H$$
 $M_i \mapsto P_i$ for all i

with the following properties:

• $\varphi(M)$ is explicitly computable for all $M \in G$

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel Recursion: compositio trees

Example: invariant

rinding reducti

Solution for leave

Recognition of the group

Verification

What is a reduction?

Let
$$G := \langle M_1, \ldots, M_k \rangle \leq \operatorname{GL}_n(\mathbb{F}_q)$$
.

A reduction is a group homomorphism

$$\varphi: G \to H$$
 $M_i \mapsto P_i$ for all i

- $\varphi(M)$ is explicitly computable for all $M \in G$
- φ is surjective: $H = \langle P_1, \dots, P_k \rangle$

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel Recursion: compositio trees

Example: invariant

Finding reduct

Solution for leave

Classifications
Recognition of the group

Verification

What is a reduction?

Let
$$G := \langle M_1, \ldots, M_k \rangle \leq \operatorname{GL}_n(\mathbb{F}_q)$$
.

A reduction is a group homomorphism

$$\varphi: G \to H$$
 $M_i \mapsto P_i$ for all i

- $\varphi(M)$ is explicitly computable for all $M \in G$
- φ is surjective: $H = \langle P_1, \dots, P_k \rangle$
- H is in some sense "smaller"
- or at least "easier to recognise constructively"

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition

trees
Example: invariant

Finding reduct

Solution for leave

Recognition of the group

Verification

What is a reduction?

Let
$$G := \langle M_1, \ldots, M_k \rangle \leq \operatorname{GL}_n(\mathbb{F}_q)$$
.

A reduction is a group homomorphism

$$\varphi: G \to H$$
 $M_i \mapsto P_i$ for all i

- $\varphi(M)$ is explicitly computable for all $M \in G$
- φ is surjective: $H = \langle P_1, \dots, P_k \rangle$
- H is in some sense "smaller"
- or at least "easier to recognise constructively"
- e.g. $H \leq S_m$ or $H \leq \operatorname{GL}_{n'}(\mathbb{F}_{q'})$ with $n' \log q' < n \log q$

Max Neunhöffer

Introduction

Matrix groups

The problen

Complexity theory
Randomised algorithm
Constructive recognition

Reduction

Homomorphisms

Computing the kernel

Recursion: composition trees

Example: invariar subspace

Finding reducti

Solution for leaves

Recognition of the group

Verification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel

Recursion: compositi trees

Example: invariant subspace

Finding reducti

Solution for leaves

Recognition of the group

Verification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: compositio

trees Example: invariant subspace

Finding reducti

Solution for leave

Recognition of the group

Verification

Computing the kernel

Let $\varphi:G\to H$ be a reduction and assume that H is already recognised constructively.

Then we can compute the kernel N of φ :

• Generate a (pseudo-) random element $M \in G$,

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm Constructive recogniti Troubles

Reduction

Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Finding reducti

Classifications

Verification

Computing the kernel

Let $\varphi:G\to H$ be a reduction and assume that H is already recognised constructively.

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel Recursion: composition trees

Example: invariant subspace

Solution for leave

Verification

Computing the kernel

Let $\varphi:G\to H$ be a reduction and assume that H is already recognised constructively.

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **3** express $\varphi(M)$ as word in the P_i ,

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel Recursion: compositio

trees
Example: invariant

Finding reducti

Solution for leave

Verification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **3** express $\varphi(M)$ as word in the P_i ,
- \bullet evaluate the same word in the M_i ,

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: compositio

trees
Example: invariant

Finding reduction

Solution for leave

Varification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **3** express $\varphi(M)$ as word in the P_i ,
- \bullet evaluate the same word in the M_i ,
- **⑤** get element M' ∈ G with $M \cdot M'^{-1} \in N$.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory Randomised algorithm Constructive recognition Troubles

Reduction

Computing the kernel
Recursion: compositio

trees
Example: invariant subspace

Finding reduction

Classifications

Recognition of the group

Verification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **3** express $\varphi(M)$ as word in the P_i ,
- \bullet evaluate the same word in the M_i ,
- **⑤** get element M' ∈ G with $M \cdot M'^{-1} \in N$.
- If M is uniformly distributed in Gthen $M \cdot M'^{-1}$ is uniformly distributed in N

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: compositio

trees
Example: invariant

Finding reduction

Classifications

Recognition of the group

Verification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **3** express $\varphi(M)$ as word in the P_i ,
- \bullet evaluate the same word in the M_i ,
- **⑤** get element M' ∈ G with $M \cdot M'^{-1} \in N$.
- If M is uniformly distributed in Gthen $M \cdot M'^{-1}$ is uniformly distributed in N
- Repeat.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

trees Example: invariant subspace

Finding reduction

Classifications
Recognition of the groups

Verification

Computing the kernel

Let $\varphi: G \to H$ be a reduction and assume that H is already recognised constructively.

Then we can compute the kernel N of φ :

- **1** Generate a (pseudo-) random element $M \in G$,
- **2** map it with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **3** express $\varphi(M)$ as word in the P_i ,
- evaluate the same word in the M_i ,
- **⑤** get element M' ∈ G with $M \cdot M'^{-1} \in N$.
- If M is uniformly distributed in Gthen $M \cdot M'^{-1}$ is uniformly distributed in N
- Repeat.

→ Monte Carlo algorithm to compute N

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Complexity theory
Randomised algorithms
Constructive recognitio

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

subspace
Finding reductions

Classifications

Verification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invariant subspace

Solution for leave
Classifications

Varification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G|=|H|\cdot |N|.$$

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recognitio
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invariant subspace

Solution for leave

Verification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

subspace Finding reduction

Solution for leave

Classifications

Varification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

Then we have recognised *G* constructively:

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

1 map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Solution for leave

Verification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

- **1** map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- \bigcirc express $\varphi(M)$ as word in the P_i ,

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

trees
Example: invariant subspace

Finding reduction

Solution for leave

Verification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

- **1** map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- \bigcirc express $\varphi(M)$ as word in the P_i ,
- \odot evaluate the same word in the M_i ,

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorith
Constructive recognit
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

trees Example: invariant subspace

Finding reduction

Solution for leaves
Classifications

Verification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

- **1** map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- \bigcirc express $\varphi(M)$ as word in the P_i ,
- \odot evaluate the same word in the M_i ,
- get element $M' \in G$ such that $M \cdot M'^{-1} \in N$,

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Solution for leave

Verification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

- **1** map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- \bigcirc express $\varphi(M)$ as word in the P_i ,
- \odot evaluate the same word in the M_i ,
- **4** get element $M' \in G$ such that $M \cdot M'^{-1} \in N$,
- **1** express $M \cdot M'^{-1}$ as word in the N_j ,

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Solution for leave

Varification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in GL_n(\mathbb{F}_q)$:

- **1** map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- \bigcirc express $\varphi(M)$ as word in the P_i ,
- \odot evaluate the same word in the M_i ,
- **9** get element $M' \in G$ such that $M \cdot M'^{-1} \in N$,
- **1** express $M \cdot M'^{-1}$ as word in the N_i ,
- **o** get M as word in the M_i and N_j : $M' = \prod \text{ in the } M_i, \quad M \cdot M'^{-1} = \prod \text{ in the } N_j$ $\Rightarrow M = (\prod \text{ in the } N_i) \cdot (\prod \text{ in the } M_i).$

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithr
Constructive recogniti
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees

subspace Finding reduction

Solution for leave
Classifications

Varification

Recognising image and kernel suffices

Let $\varphi: G \to H$ be a reduction and assume that both H and the kernel $N = \langle N_1, \dots, N_m \rangle$ of φ are already recognised constructively.

$$|G| = |H| \cdot |N|$$
. And for $M \in \mathrm{GL}_n(\mathbb{F}_q)$:

- **1** map M with φ onto $\varphi(M) \in H = \langle P_1, \dots, P_k \rangle$,
- **2** express $\varphi(M)$ as word in the P_i ,
- \odot evaluate the same word in the M_i ,
- **9** get element $M' \in G$ such that $M \cdot M'^{-1} \in N$,
- \odot express $M \cdot M'^{-1}$ as word in the N_i ,
- **3** get M as word in the M_i and N_j : $M' = \prod \text{ in the } M_i, \quad M \cdot M'^{-1} = \prod \text{ in the } N_j$ $⇒ M = (\prod \text{ in the } N_i) \cdot (\prod \text{ in the } M_i).$
- If $M \notin G$, then at least one step does not work.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The probler

Randomised algorithm
Constructive recognition

Reduction

Homomorphism

Computing the kerne

Recursion: composition

Example: invariar subspace

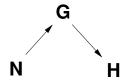
Finding reduction

Solution for leave

Classifications

Verification

Recursion: composition trees We get a tree:



Up arrows: inclusions

Down arrows: homomorphisms

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The probler

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invariar subspace

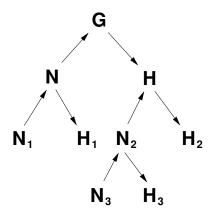
Finding reduction

Solution for leave

Recognition of the group

Varification

Recursion: composition trees We get a tree:



Up arrows: inclusions

Down arrows: homomorphisms

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

subspace Finding reduction

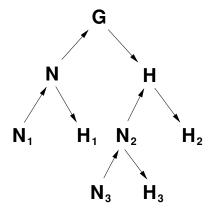
Finding reducti

Solution for leave

ricoognition or the

Recursion: composition trees

We get a tree:



Up arrows: inclusions

Down arrows: homomorphisms

Old idea, substantial improvements are still being made

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition

Reduction

Homomorphisms

Computing the kernel
Recursion: composition

Example: invariant subspace

Finding reduction

Solution for leaves

Recognition of the group

Verification

Example: invariant subspace

Let $V = \mathbb{F}_q^n$, then G acts on V. Let $W \le V$ be an invariant subspace, i.e.:

MW = W for all $M \in G$

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel

Recursion: composition
trees

trees
Example: invariant

Finding reductio

Solution for leaves

Classifications

Verification

Example: invariant subspace

Let $V = \mathbb{F}_q^n$, then G acts on V. Let $W \leq V$ be an invariant subspace, i.e.:

$$MW = W$$
 for all $M \in G$

Choose basis (w_1, \ldots, w_d) of W and extend to a basis

$$(w_1,\ldots,w_d,w_{d+1},\ldots,w_n)$$

of V. After a base change the matrices in G look like this:

$$\begin{vmatrix} A & B \\ \hline \mathbf{0} & D \end{vmatrix} \quad \text{with } A \in \mathbb{F}_q^{d \times d}, B \in \mathbb{F}_q^{d \times (n-d)}, D \in \mathbb{F}_q^{(n-d) \times (n-d)}$$

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees

Example: invariant subspace

Finding reduction

Solution for leaves

Verification

Example: invariant subspace

Let $V = \mathbb{F}_q^n$, then G acts on V.

Let $W \leq V$ be an invariant subspace, i.e.:

$$MW = W$$
 for all $M \in G$

Choose basis (w_1, \ldots, w_d) of W and extend to a basis

$$(w_1,\ldots,w_d,w_{d+1},\ldots,w_n)$$

of V. After a base change the matrices in G look like this:

$$\begin{bmatrix} A & B \\ \hline \mathbf{0} & D \end{bmatrix} \quad \text{with } A \in \mathbb{F}_q^{d \times d}, B \in \mathbb{F}_q^{d \times (n-d)}, D \in \mathbb{F}_q^{(n-d) \times (n-d)}$$

and

$$G o \operatorname{GL}_{n-d}(\mathbb{F}_q), \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \mapsto D$$

is a homomorphism of groups.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition

Reduction

Homomorphisms

Computing the kernel Recursion: compositio

Example: invariant subspace

Finding reduction

Solution for leaves

Classifications Becognition of the group

Verification

Example: invariant subspace

$$G o \operatorname{GL}_{n-d}(\mathbb{F}_q), \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \mapsto D$$

is a homomorphism of groups, its kernel is

$$N := \left\{ \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \in G \mid D = \mathbf{1}
ight\}.$$

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms

Computing the kernel

Recursion: composition

Example: invariant subspace

Finding reduction

Solution for leaves

Recognition of the group

Verification

Example: invariant subspace

$$G o \operatorname{GL}_{n-d}(\mathbb{F}_q), \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \mapsto D$$

is a homomorphism of groups, its kernel is

$$N := \left\{ \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \in G \mid D = \mathbf{1}
ight\}.$$

The mapping

$$N \to \mathrm{GL}_d(\mathbb{F}_q), \left| \begin{array}{cc} A & B \\ \mathbf{0} & \mathbf{1} \end{array} \right| \mapsto A$$

also is a homomorphism of groups and has kernel

$$N_2 := \left\{ \left[egin{array}{cc} A & B \\ \mathbf{0} & D \end{array}
ight] \in G \mid A = D = \mathbf{1}
ight\}.$$

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition

Example: invariant subspace

Finding reducti

Solution for leaves

ecognition of the grou

Verification

Example: invariant subspace

$$G o \operatorname{GL}_{n-d}(\mathbb{F}_q), \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \mapsto D$$

is a homomorphism of groups, its kernel is

$$N := \left\{ \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \in G \mid D = \mathbf{1}
ight\}.$$

The mapping

$$N \to \mathrm{GL}_d(\mathbb{F}_q), \left| \begin{array}{cc} A & B \\ \mathbf{0} & \mathbf{1} \end{array} \right| \mapsto A$$

also is a homomorphism of groups and has kernel

$$N_2 := \left\{ \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \in G \mid A = D = \mathbf{1}
ight\}.$$

This group is a *p*-group for $q = p^e$:

$$\left[\begin{array}{cc} \mathbf{1} & B \\ \mathbf{0} & \mathbf{1} \end{array}\right] \cdot \left[\begin{array}{cc} \mathbf{1} & B' \\ \mathbf{0} & \mathbf{1} \end{array}\right] = \left[\begin{array}{cc} \mathbf{1} & B + B' \\ \mathbf{0} & \mathbf{1} \end{array}\right]$$

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithm
Constructive recogniti
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition

trees Example: invariant subspace

Finding reduct

Solution for leaves

Classifications
Recognition of the groups

Verification

Example: invariant subspace

$$G o \operatorname{GL}_{n-d}(\mathbb{F}_q), \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \mapsto D$$

is a homomorphism of groups, its kernel is

$$N := \left\{ \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \in G \mid D = \mathbf{1}
ight\}.$$

The mapping

$$N o \operatorname{GL}_d(\mathbb{F}_q), \left[egin{array}{cc} A & B \ \mathbf{0} & \mathbf{1} \end{array}
ight] \mapsto A$$

also is a homomorphism of groups and has kernel

$$N_2 := \left\{ \left[egin{array}{cc} A & B \ \mathbf{0} & D \end{array}
ight] \in G \mid A = D = \mathbf{1}
ight\}.$$

This group is a *p*-group for $q = p^e$:

$$\left[\begin{array}{cc} \mathbf{1} & B \\ \mathbf{0} & \mathbf{1} \end{array}\right] \cdot \left[\begin{array}{cc} \mathbf{1} & B' \\ \mathbf{0} & \mathbf{1} \end{array}\right] = \left[\begin{array}{cc} \mathbf{1} & B + B' \\ \mathbf{0} & \mathbf{1} \end{array}\right]$$

Together with a reduction additional information is gained!

Max Neunhöffer

Reduction

Finding reductions

How to find reductions?

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The probler

Randomised algorithm
Constructive recognition

Reduction

Homomorphisms

Computing the kernel Recursion: composition trees

Finding reductions

Finding reductio

Solution for leaves

lassifications lecognition of the group

Verification

How to find reductions?

Aschbacher has defined classes C1 to C8 of subgroups of $GL_n(\mathbb{F}_q)$.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithms
Constructive recognitio
Troubles

Reduction

Computing the kernel Recursion: composition trees Example: invariant

Finding reductions

Solution for leave

Recognition of the group

Verification

How to find reductions?

Aschbacher has defined classes C1 to C8 of subgroups of $GL_n(\mathbb{F}_q)$.

Theorem (Aschbacher, 1984)

Let $G \leq \operatorname{GL}_n(\mathbb{F}_q)$ and $Z := G \cap Z(\operatorname{GL}_n(\mathbb{F}_q))$ the subgroup of scalar matrices. Then G lies in at least one of the classes C1 to C8 or we have:

- $T \subseteq G/Z \subseteq Aut(T)$ for a non-abelian simple group T, and
- G acts absolutely irreducibly on $V = \mathbb{F}_q^n$.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithms
Constructive recognitio
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reductions

Solution for leave

Recognition of the group

Verification

How to find reductions?

Aschbacher has defined classes C1 to C8 of subgroups of $GL_n(\mathbb{F}_q)$.

Theorem (Aschbacher, 1984)

Let $G \leq \operatorname{GL}_n(\mathbb{F}_q)$ and $Z := G \cap Z(\operatorname{GL}_n(\mathbb{F}_q))$ the subgroup of scalar matrices. Then G lies in at least one of the classes C1 to C8 or we have:

- $T \subseteq G/Z \subseteq Aut(T)$ for a non-abelian simple group T, and
- G acts absolutely irreducibly on $V = \mathbb{F}_q^n$.

(This last case is called C9.)

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithms
Constructive recognitio
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reductions

Solution for leave

Verification

How to find reductions?

Aschbacher has defined classes C1 to C8 of subgroups of $GL_n(\mathbb{F}_q)$.

Theorem (Aschbacher, 1984)

Let $G \leq \operatorname{GL}_n(\mathbb{F}_q)$ and $Z := G \cap Z(\operatorname{GL}_n(\mathbb{F}_q))$ the subgroup of scalar matrices. Then G lies in at least one of the classes C1 to C8 or we have:

- $T \subseteq G/Z \subseteq Aut(T)$ for a non-abelian simple group T, and
- G acts absolutely irreducibly on $V = \mathbb{F}_q^n$.

(This last case is called C9.)

Thus we can call in heavy artillery:

the classification of finite simple groups

Max Neunhöffer

Introduction

Matrix groups
Constructive recognition

The problem

Randomised algorithms
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reductions

Solution for leave

Verification

How to find reductions?

Aschbacher has defined classes C1 to C8 of subgroups of $GL_n(\mathbb{F}_q)$.

Theorem (Aschbacher, 1984)

Let $G \leq \operatorname{GL}_n(\mathbb{F}_q)$ and $Z := G \cap Z(\operatorname{GL}_n(\mathbb{F}_q))$ the subgroup of scalar matrices. Then G lies in at least one of the classes C1 to C8 or we have:

- $T \subseteq G/Z \subseteq Aut(T)$ for a non-abelian simple group T, and
- G acts absolutely irreducibly on $V = \mathbb{F}_q^n$.

(This last case is called C9.)

Thus we can call in heavy artillery:

- the classification of finite simple groups
- the modular representation theory of simple groups

Max Neunhöffer

Introduction

Matrix groups

The proble

Randomised algorithm
Constructive recognition

Reduction

Homomorphisms

Computing the kernel Recursion: compositio trees

Example: invarian subspace

Finding reduction

Solution for leaves

Classifications

Verification

Approach for leaves of the tree

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leave: Classifications

Recognition of the group

Verification

Approach for leaves of the tree

If none of the algorithms for C1 to C8 has succeeded:

For "small" groups compute direct isomorphism onto a permutation group.

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Complexity theory
Randomised algorithm
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leaves

Recognition of the group

Approach for leaves of the tree

- For "small" groups compute direct isomorphism onto a permutation group.
- **2** Determine, for which (simple) group $T \leq G/Z \leq \operatorname{Aut}(T)$ holds.

Max Neunhöffer

Introduction

Matrix groups
Constructive recognitio

The problem

Randomised algorithm
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leave Classifications

Approach for leaves of the tree

- For "small" groups compute direct isomorphism onto a permutation group.
- 2 Determine, for which (simple) group $T \leq G/Z \leq \operatorname{Aut}(T)$ holds.
- Find an explicit isomorphism onto a "standard copy" of an intermediate group S.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithm Constructive recogniti Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace
Finding reductions

Solution for leave Classifications

Varification

Approach for leaves of the tree

- For "small" groups compute direct isomorphism onto a permutation group.
- Determine, for which (simple) group $T \le G/Z \le \operatorname{Aut}(T)$ holds.
- Find an explicit isomorphism onto a "standard copy" of an intermediate group S.
- Finally use information about S to recognise G constructively.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Complexity theory
Randomised algorithn
Constructive recogniti
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace
Finding reductions

Solution for leave

Verificatio

Approach for leaves of the tree

If none of the algorithms for C1 to C8 has succeeded:

- For "small" groups compute direct isomorphism onto a permutation group.
- Determine, for which (simple) group $T \le G/Z \le \operatorname{Aut}(T)$ holds.
- Find an explicit isomorphism onto a "standard copy" of an intermediate group S.
- Finally use information about S to recognise G constructively.

This uses:

- the classification of finite simple groups
- information about their automorphism groups
- information about element orders
- information about conjugacy classes
- classifications of the irreducible representations
- information about the subgroup structure

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leaves

Recognition of the groups

Verification

Non-constructive recognition

Methods for non-constructive recognition:

Knowledge about representations narrows down the possibilities

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recogniti
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace
Finding reductions

Solution for leave: Classifications

Recognition of the groups

Verification

Non-constructive recognition

Methods for non-constructive recognition:

- Knowledge about representations narrows down the possibilities
- Statistics about orders of random elements

Max Neunhöffer

Introduction

Matrix groups

Constructive recognitio

The problem

Randomised algorithr
Constructive recogniti
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leave

Classifications
Recognition of the groups

Verification

Non-constructive recognition

Methods for non-constructive recognition:

- Knowledge about representations narrows down the possibilities
- Statistics about orders of random elements

Usually this leads to Monte Carlo algorithms.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leaves

Recognition of the group

Verification

Verification

Everywhere we used randomised methods: Las Vegas and Monte Carlo.

⇒ We have to check whether our result is correct!

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant

Finding reduction

Classifications

Verification

Verification

Everywhere we used randomised methods: Las Vegas and Monte Carlo.

⇒ We have to check whether our result is correct!

Idea:

Find (short) presentations for the leaf-groups,

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Homomorphisms
Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Finding reduction

Solution for leaves
Classifications

Verification

Verification

Everywhere we used randomised methods: Las Vegas and Monte Carlo.

⇒ We have to check whether our result is correct!

Idea:

- Find (short) presentations for the leaf-groups,
- put these together to one for the whole group.

Max Neunhöffer

Introduction

Matrix groups

Constructive recognition

The problem

Randomised algorithm
Constructive recognition
Troubles

Reduction

Computing the kernel
Recursion: composition
trees
Example: invariant
subspace

Solution for leave

Verification

Verification

Everywhere we used randomised methods: Las Vegas and Monte Carlo.

⇒ We have to check whether our result is correct!

Idea:

- Find (short) presentations for the leaf-groups,
- put these together to one for the whole group.
- Check the relations and thus prove the result.