Polynomial-time algorithms for algebras with involution

Peter Brooksbank

Bucknell University

The 3rd Workshop on Algebra, Algorithms & Applications

De Brún Centre for Computational Algebra

National University of Ireland, Galway (30 November–10 December, 2009)

Overview

1. Polynomial-time theory of matrix algebras over finite fields.
2. Extending the theory to matrix $*$-algebras.
3. An application to computation with p-groups.

(joint work with James Wilson)

Overview

1. Polynomial-time theory of matrix algebras over finite fields.
2. Extending the theory to matrix ∗-algebras.
3. An application to computation with p-groups.

(joint work with James Wilson)

Overview

1. Polynomial-time theory of matrix algebras over finite fields.
2. Extending the theory to matrix $*$-algebras.
3. An application to computation with p-groups.

(joint work with James Wilson)

Overview

1. Polynomial-time theory of matrix algebras over finite fields.
2. Extending the theory to matrix $*$-algebras.
3. An application to computation with p-groups.

(joint work with James Wilson)

Overview

1. Polynomial-time theory of matrix algebras over finite fields.
2. Extending the theory to matrix $*$-algebras.
3. An application to computation with p-groups.

(joint work with James Wilson)

Wedderburn-Malcev Principal Theorem

$\mathbb{F}_q$ = field of q elements
V = finite-dimensional $\mathbb{F}_q$-space
$\mathrm{End}_{\mathbb{F}_q}(V)$ = algebra of $\mathbb{F}_q$-linear endomorphisms of V

*Let $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, and let $J(A)$ be the **Jacobson radical** of A.*

1. *There is a subalgebra S of A such that $A = J(A) \oplus S$.*
2. *If $T \leqslant A$ with $A = J(A) \oplus T$ then $\exists u \in J(A)$ such that $(1+u)^{-1}T(1+u) = S$.*
3. *There is a collection $\{S_i : 1 \leqslant i \leqslant n\}$ of minimal ideals of S such that $S = S_1 \oplus \ldots \oplus S_n$.*
4. *For each $1 \leqslant i \leqslant n$ there is an extension $\mathbb{K}_i$ of $\mathbb{F}_q$ and a $\mathbb{K}_i$-space W_i such that $S_i \cong \mathrm{End}_{\mathbb{K}_i}(W_i)$.*

Wedderburn-Malcev Principal Theorem

$\mathbb{F}_q$ = field of q elements
V = finite-dimensional $\mathbb{F}_q$-space
$\mathrm{End}_{\mathbb{F}_q}(V)$ = algebra of $\mathbb{F}_q$-linear endomorphisms of V

*Let $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, and let $J(A)$ be the **Jacobson radical** of A.*

1. *There is a subalgebra S of A such that $A = J(A) \oplus S$.*
2. *If $T \leqslant A$ with $A = J(A) \oplus T$ then $\exists u \in J(A)$ such that $(1+u)^{-1}T(1+u) = S$.*
3. *There is a collection $\{S_i : 1 \leqslant i \leqslant n\}$ of minimal ideals of S such that $S = S_1 \oplus \ldots \oplus S_n$.*
4. *For each $1 \leqslant i \leqslant n$ there is an extension $\mathbb{K}_i$ of $\mathbb{F}_q$ and a $\mathbb{K}_i$-space W_i such that $S_i \cong \mathrm{End}_{\mathbb{K}_i}(W_i)$.*

Algorithmic preliminaries

- Assume that a matrix algebra is given as the **enveloping algebra** of some set $X \subset \mathrm{End}_{\mathbb{F}_q}(V)$, where $V = {\mathbb{F}_q}^d$.
- The input length of $\mathrm{Env}(X)$ is roughly $|X|d^2 \log q$.
- Use the **arithmetic model**: algorithmic complexity is the total number of operations in $\mathbb{F}_q$.
- Assume an algorithm that computes independent, (nearly) uniformly distributed random elements of $\mathrm{Env}(X)$. Denote its complexity by ξ.
- Algorithms are randomized, and of the **Las Vegas** variety.
- **Polynomial-time algorithm**: Algorithm having complexity that is polynomial in $|X|d^2 \log q$.

Algorithmic preliminaries

- Assume that a matrix algebra is given as the **enveloping algebra** of some set $X \subset \mathrm{End}_{\mathbb{F}_q}(V)$, where $V = {\mathbb{F}_q}^d$.
- The input length of $\mathrm{Env}(X)$ is roughly $|X|d^2 \log q$.
- Use the **arithmetic model**: algorithmic complexity is the total number of operations in $\mathbb{F}_q$.
- Assume an algorithm that computes independent, (nearly) uniformly distributed random elements of $\mathrm{Env}(X)$. Denote its complexity by ξ.
- Algorithms are randomized, and of the **Las Vegas** variety.
- **Polynomial-time algorithm**: Algorithm having complexity that is polynomial in $|X|d^2 \log q$.

Algorithmic preliminaries

- Assume that a matrix algebra is given as the **enveloping algebra** of some set $X \subset \mathrm{End}_{\mathbb{F}_q}(V)$, where $V = {\mathbb{F}_q}^d$.
- The input length of $\mathrm{Env}(X)$ is roughly $|X|d^2 \log q$.
- Use the **arithmetic model**: algorithmic complexity is the total number of operations in $\mathbb{F}_q$.
- Assume an algorithm that computes independent, (nearly) uniformly distributed random elements of $\mathrm{Env}(X)$. Denote its complexity by ξ.
- Algorithms are randomized, and of the **Las Vegas** variety.
- **Polynomial-time algorithm**: Algorithm having complexity that is polynomial in $|X|d^2 \log q$.

Algorithmic preliminaries

- Assume that a matrix algebra is given as the **enveloping algebra** of some set $X \subset \mathrm{End}_{\mathbb{F}_q}(V)$, where $V = {\mathbb{F}_q}^d$.
- The input length of $\mathrm{Env}(X)$ is roughly $|X|d^2 \log q$.
- Use the **arithmetic model**: algorithmic complexity is the total number of operations in $\mathbb{F}_q$.
- Assume an algorithm that computes independent, (nearly) uniformly distributed random elements of $\mathrm{Env}(X)$. Denote its complexity by ξ.
- Algorithms are randomized, and of the **Las Vegas** variety.
- **Polynomial-time algorithm**: Algorithm having complexity that is polynomial in $|X|d^2 \log q$.

Algorithmic preliminaries

- Assume that a matrix algebra is given as the **enveloping algebra** of some set $X \subset \mathrm{End}_{\mathbb{F}_q}(V)$, where $V = {\mathbb{F}_q}^d$.
- The input length of $\mathrm{Env}(X)$ is roughly $|X|d^2 \log q$.
- Use the **arithmetic model**: algorithmic complexity is the total number of operations in $\mathbb{F}_q$.
- Assume an algorithm that computes independent, (nearly) uniformly distributed random elements of $\mathrm{Env}(X)$. Denote its complexity by ξ.
- Algorithms are randomized, and of the **Las Vegas** variety.
- **Polynomial-time algorithm**: Algorithm having complexity that is polynomial in $|X|d^2 \log q$.

Algorithmic preliminaries

- Assume that a matrix algebra is given as the **enveloping algebra** of some set $X \subset \mathrm{End}_{\mathbb{F}_q}(V)$, where $V = {\mathbb{F}_q}^d$.
- The input length of $\mathrm{Env}(X)$ is roughly $|X|d^2 \log q$.
- Use the **arithmetic model**: algorithmic complexity is the total number of operations in $\mathbb{F}_q$.
- Assume an algorithm that computes independent, (nearly) uniformly distributed random elements of $\mathrm{Env}(X)$. Denote its complexity by ξ.
- Algorithms are randomized, and of the **Las Vegas** variety.
- **Polynomial-time algorithm**: Algorithm having complexity that is polynomial in $|X|d^2 \log q$.

Polynomial-time algorithms for matrix algebras

Theorem (Ivanyos, 2000)

There is a polynomial-time Las Vegas algorithm which, given any algebra $A = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$*, finds the following:*

1. *The Jacobson radical,* $J(A)$*, of* A*.*
2. *A subring* S *of* A *such that* $A = J(A) \oplus S$*.*

Theorem (Eberly & Giesbrecht, 2000)

There is a polynomial-time Las Vegas algorithm which, given a ***semisimple*** *algebra* $S = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$*, finds the following:*

1. *Minimal ideals* $S_1, \ldots, S_n$ *of* S *such that* $S = S_1 \oplus \ldots \oplus S_n$*.*
2. *For each* $1 \leqslant i \leqslant n$*, an extension* $\mathbb{K}_i$ *of* $\mathbb{F}_q$ *and a* $\mathbb{K}_i$*-space* W_i *such that* $A_i \cong \mathrm{End}_{\mathbb{K}_i}(W_i)$*.*

The complexity of both algorithms is $O^{\sim}(|X|(d^3 + d^2 \log|\mathbb{F}_q| + \xi))$.

Polynomial-time algorithms for matrix algebras

Theorem (Ivanyos, 2000)

There is a polynomial-time Las Vegas algorithm which, given any algebra $A = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$*, finds the following:*

1. *The Jacobson radical,* $J(A)$*, of* A*.*
2. *A subring* S *of* A *such that* $A = J(A) \oplus S$*.*

Theorem (Eberly & Giesbrecht, 2000)

There is a polynomial-time Las Vegas algorithm which, given a ***semisimple*** *algebra* $S = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$*, finds the following:*

1. *Minimal ideals* $S_1, \ldots, S_n$ *of* S *such that* $S = S_1 \oplus \ldots \oplus S_n$*.*
2. *For each* $1 \leqslant i \leqslant n$*, an extension* $\mathbb{K}_i$ *of* $\mathbb{F}_q$ *and a* $\mathbb{K}_i$*-space* W_i *such that* $A_i \cong \mathrm{End}_{\mathbb{K}_i}(W_i)$*.*

The complexity of both algorithms is $O^{\sim}(|X|(d^3 + d^2 \log|\mathbb{F}_q| + \xi))$.

Polynomial-time algorithms for matrix algebras

Theorem (Ivanyos, 2000)

There is a polynomial-time Las Vegas algorithm which, given any algebra $A = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, *finds the following:*

1. *The Jacobson radical,* $J(A)$, *of* A.
2. *A subring* S *of* A *such that* $A = J(A) \oplus S$.

Theorem (Eberly & Giesbrecht, 2000)

There is a polynomial-time Las Vegas algorithm which, given a ***semisimple*** *algebra* $S = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, *finds the following:*

1. *Minimal ideals* $S_1, \ldots, S_n$ *of* S *such that* $S = S_1 \oplus \ldots \oplus S_n$.
2. *For each* $1 \leqslant i \leqslant n$, *an extension* $\mathbb{K}_i$ *of* $\mathbb{F}_q$ *and a* $\mathbb{K}_i$*-space* W_i *such that* $A_i \cong \mathrm{End}_{\mathbb{K}_i}(W_i)$.

The complexity of both algorithms is $O^{\sim}(|X|(d^3 + d^2 \log|\mathbb{F}_q| + \xi))$.

Constructing the unit group of a matrix algebra

For a ring R, let

$$U(R) = \{x \in R : x \text{ is a unit}\}.$$

Given $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, find $U(A) = A \cap \mathrm{GL}_{\mathbb{F}_q}(V)$ as follows:

1. Write $A = J(A) \oplus (S_1 \oplus \ldots \oplus S_n)$.
2. Set Y := basis of $J(A)$, set $Q := \langle 1 + u : u \in Y \rangle$.
3. For $1 \leqslant i \leqslant n$, if $S_i \cong \mathrm{End}_{K_i}(W_i)$, then $U(S_i) \cong \mathrm{GL}_{K_i}(W_i)$.
4. Return $U(A) = Q \rtimes (U(S_1) \times \ldots \times U(S_n))$.

Constructing the unit group of a matrix algebra

For a ring R, let

$$U(R) = \{x \in R : x \text{ is a unit}\}.$$

Given $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, find $U(A) = A \cap \mathrm{GL}_{\mathbb{F}_q}(V)$ as follows:

1. Write $A = J(A) \oplus (S_1 \oplus \ldots \oplus S_n)$.
2. Set $Y :=$ basis of $J(A)$, set $Q := \langle 1 + u : u \in Y \rangle$.
3. For $1 \leqslant i \leqslant n$, if $S_i \cong \mathrm{End}_{K_i}(W_i)$, then $U(S_i) \cong \mathrm{GL}_{K_i}(W_i)$.
4. Return $U(A) = Q \rtimes (U(S_1) \times \ldots \times U(S_n))$.

Constructing the unit group of a matrix algebra

For a ring R, let

$$U(R) = \{x \in R : x \text{ is a unit}\}.$$

Given $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, find $U(A) = A \cap \mathrm{GL}_{\mathbb{F}_q}(V)$ as follows:

1. Write $A = J(A) \oplus (S_1 \oplus \ldots \oplus S_n)$.
2. Set $Y :=$ basis of $J(A)$, set $Q := \langle 1 + u : u \in Y \rangle$.
3. For $1 \leqslant i \leqslant n$, if $S_i \cong \mathrm{End}_{K_i}(W_i)$, then $U(S_i) \cong \mathrm{GL}_{K_i}(W_i)$.
4. Return $U(A) = Q \rtimes (U(S_1) \times \ldots \times U(S_n))$.

Constructing the unit group of a matrix algebra

For a ring R, let

$$U(R) = \{x \in R : x \text{ is a unit}\}.$$

Given $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, find $U(A) = A \cap \mathrm{GL}_{\mathbb{F}_q}(V)$ as follows:

1. Write $A = J(A) \oplus (S_1 \oplus \ldots \oplus S_n)$.
2. Set Y := basis of $J(A)$, set $Q := \langle 1 + u : u \in Y \rangle$.
3. For $1 \leqslant i \leqslant n$, if $S_i \cong \mathrm{End}_{K_i}(W_i)$, then $U(S_i) \cong \mathrm{GL}_{K_i}(W_i)$.
4. Return $U(A) = Q \rtimes (U(S_1) \times \ldots \times U(S_n))$.

Constructing the unit group of a matrix algebra

For a ring R, let

$$U(R) = \{x \in R : x \text{ is a unit}\}.$$

Given $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, find $U(A) = A \cap \mathrm{GL}_{\mathbb{F}_q}(V)$ as follows:

1. Write $A = J(A) \oplus (S_1 \oplus \ldots \oplus S_n)$.
2. Set Y := basis of $J(A)$, set $Q := \langle 1 + u : u \in Y \rangle$.
3. For $1 \leqslant i \leqslant n$, if $S_i \cong \mathrm{End}_{K_i}(W_i)$, then $U(S_i) \cong \mathrm{GL}_{K_i}(W_i)$.
4. Return $U(A) = Q \rtimes (U(S_1) \times \ldots \times U(S_n))$.

Constructing the unit group of a matrix algebra

For a ring R, let

$$U(R) = \{x \in R : x \text{ is a unit}\}.$$

Given $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, find $U(A) = A \cap \mathrm{GL}_{\mathbb{F}_q}(V)$ as follows:

1. Write $A = J(A) \oplus (S_1 \oplus \ldots \oplus S_n)$.
2. Set Y := basis of $J(A)$, set $Q := \langle 1 + u : u \in Y \rangle$.
3. For $1 \leqslant i \leqslant n$, if $S_i \cong \mathrm{End}_{K_i}(W_i)$, then $U(S_i) \cong \mathrm{GL}_{K_i}(W_i)$.
4. Return $U(A) = Q \rtimes (U(S_1) \times \ldots \times U(S_n))$.

Algebras with involution

A **$*$-algebra** is an associative, unital algebra A equipped with an **involution** (anti-automorphism of order 2) which is denoted $*$.

In algorithms we assume a procedure (or oracle) that effects the involution on A, and denote its complexity by $\mathrm{Star}(A, *)$.

In practice the involution is defined just on the generators of A.

Algebras with involution

A $*$**-algebra** is an associative, unital algebra A equipped with an **involution** (anti-automorphism of order 2) which is denoted $*$.

In algorithms we assume a procedure (or oracle) that effects the involution on A, and denote its complexity by $\mathrm{Star}(A, *)$.

In practice the involution is defined just on the generators of A.

Algebras with involution

A $*$**-algebra** is an associative, unital algebra A equipped with an **involution** (anti-automorphism of order 2) which is denoted $*$.

In algorithms we assume a procedure (or oracle) that effects the involution on A, and denote its complexity by $\mathrm{Star}(A, *)$.

In practice the involution is defined just on the generators of A.

Group algebras

G = finite group
$\mathbb{K}$ = any field
$\mathbb{K}G$ group algebra over $\mathbb{K}$.

Inversion extends $\mathbb{K}$-linearly to an anti-automorphism of $\mathbb{K}G$:

$$\left(\sum_{g\in G} \alpha_g g\right)^* := \sum_{g\in G} \alpha_g g^{-1}$$

Thus $\mathbb{K}G$ admits a natural involution.

Group algebras

G = finite group
$\mathbb{K}$ = any field
$\mathbb{K}G$ group algebra over $\mathbb{K}$.

Inversion extends $\mathbb{K}$-linearly to an anti-automorphism of $\mathbb{K}G$:

$$\left(\sum_{g\in G} \alpha_g g\right)^* := \sum_{g\in G} \alpha_g g^{-1}$$

Thus $\mathbb{K}G$ admits a natural involution.

Algebras of adjoints

V, W $= \mathbb{K}$-spaces
E $=$ $\mathrm{End}_{\mathbb{K}}(V)$
$b\colon V \times V \to W$ a $\mathbb{K}$-bilinear map.
Define

$$\mathrm{Adj}(b) = \{(f,g) \in E \times E^{\mathrm{op}} : b(uf, v) = b(u, vg) \ \ \forall u, v \in V\}$$

the **algebra of adjoints** of b.

- b nondegenerate and $(f,g) \in \mathrm{Adj}(b) \Rightarrow g$ is determined by f.
- b **Hermitian**[1] and $(f,g) \in \mathrm{Adj}(b) \Rightarrow (g,f) \in \mathrm{Adj}(b)$.

Restricting to the first coordinate, $f^* := g$ is an involution on $\mathrm{Adj}(b)$.

[1] $\exists \theta \in \mathrm{GL}(W)$ such that $b(u,v) = b(v,u)\theta$ for all $u, v \in V$

Algebras of adjoints

V, W $= \mathbb{K}$-spaces
$E = \mathrm{End}_{\mathbb{K}}(V)$
$b\colon V \times V \to W$ a $\mathbb{K}$-bilinear map.
Define

$$\mathrm{Adj}(b) = \{(f,g) \in E \times E^{\mathrm{op}} : b(uf, v) = b(u, vg) \ \ \forall u, v \in V\}$$

the **algebra of adjoints** of b.

- b nondegenerate and $(f,g) \in \mathrm{Adj}(b) \Rightarrow g$ is determined by f.
- b **Hermitian**[1] and $(f,g) \in \mathrm{Adj}(b) \Rightarrow (g,f) \in \mathrm{Adj}(b)$.

Restricting to the first coordinate, $f^* := g$ is an involution on $\mathrm{Adj}(b)$.

[1] $\exists \theta \in \mathrm{GL}(W)$ such that $b(u,v) = b(v,u)\theta$ for all $u, v \in V$

Algebras of adjoints

V, W $= \mathbb{K}$-spaces
E $=$ $\mathrm{End}_{\mathbb{K}}(V)$
$b\colon V \times V \to W$ a $\mathbb{K}$-bilinear map.
Define

$$\mathrm{Adj}(b) = \{(f,g) \in E \times E^{\mathrm{op}} : b(uf, v) = b(u, vg) \ \ \forall u, v \in V\}$$

the **algebra of adjoints** of b.

- b nondegenerate and $(f,g) \in \mathrm{Adj}(b) \Rightarrow g$ is determined by f.
- b **Hermitian**[1] and $(f,g) \in \mathrm{Adj}(b) \Rightarrow (g,f) \in \mathrm{Adj}(b)$.

Restricting to the first coordinate, $f^* := g$ is an involution on $\mathrm{Adj}(b)$.

[1] $\exists \theta \in \mathrm{GL}(W)$ such that $b(u,v) = b(v,u)\theta$ for all $u, v \in V$

Algebras of adjoints

V, W $= \mathbb{K}$-spaces
E $=$ $\mathrm{End}_{\mathbb{K}}(V)$
$b\colon V \times V \to W$ a $\mathbb{K}$-bilinear map.
Define

$$\mathrm{Adj}(b) = \{(f,g) \in E \times E^{\mathrm{op}} : b(uf,v) = b(u,vg) \ \ \forall u,v \in V\}$$

the **algebra of adjoints** of b.

- b nondegenerate and $(f,g) \in \mathrm{Adj}(b) \Rightarrow g$ is determined by f.
- b **Hermitian**[1] and $(f,g) \in \mathrm{Adj}(b) \Rightarrow (g,f) \in \mathrm{Adj}(b)$.

Restricting to the first coordinate, $f^* := g$ is an involution on $\mathrm{Adj}(b)$.

[1] $\exists \theta \in \mathrm{GL}(W)$ such that $b(u,v) = b(v,u)\theta$ for all $u,v \in V$.

Algebras of adjoints

V, W $= \mathbb{K}$-spaces
E $=$ $\mathrm{End}_{\mathbb{K}}(V)$
$b\colon V \times V \to W$ a $\mathbb{K}$-bilinear map.
Define

$$\mathrm{Adj}(b) = \{(f,g) \in E \times E^{\mathrm{op}} : b(uf,v) = b(u,vg) \ \ \forall u,v \in V\}$$

the **algebra of adjoints** of b.

- b nondegenerate and $(f,g) \in \mathrm{Adj}(b) \Rightarrow g$ is determined by f.
- b **Hermitian**[1] and $(f,g) \in \mathrm{Adj}(b) \Rightarrow (g,f) \in \mathrm{Adj}(b)$.

Restricting to the first coordinate, $f^* := g$ is an involution on $\mathrm{Adj}(b)$.

[1] $\exists \theta \in \mathrm{GL}(W)$ such that $b(u,v) = b(v,u)\theta$ for all $u, v \in V$.

Structures for $*$-rings

The category of $*$-rings has all the usual structures:

- **$*$-homomorphism** $(R, \varphi) \to (S, \bullet)$: a ring homomorphism $\varphi\colon R \to S$ such that $r^*\varphi = r\varphi^\bullet$ for all $r \in R$.
- **$*$-ideal of** $(R, *)$: a $*$-invariant ideal of R.
- **simple $*$-ring**: a $*$-ring having no proper $*$-ideals.

Radical-semisimple structure of ∗-algebras

Theorem (Albert, 1961)

The following hold for a ∗-algebra A:

1. $J(A)$ *is a ∗-ideal of A.*
2. $A/J(A)$ *is ∗-isomorphic to* $(T_1, *) \oplus \ldots \oplus (T_m, *)$*, where* $(T_i, *)$ *is a simple ∗-algebra.*

Classification of simple ∗-algebras

Theorem (Albert, Jacobson)

A finite simple ∗-algebra is ∗-isomorphic to one of the following types, for some n and q:

1. *Orthogonal:* $\mathrm{O}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto DX^{\mathrm{tr}}D^{-1})$
2. *Symplectic:* $\mathrm{S}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto JX^{\mathrm{tr}}J^{-1})$
3. *Unitary:* $\mathrm{U}(n, \mathbb{F}_{q^2}) = (M_n(\mathbb{F}_{q^2}), X \mapsto \overline{X}^{\mathrm{tr}})$
4. *Exchange:* $\mathrm{X}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q) \oplus M_n(\mathbb{F}_q), (X, Y) \mapsto (Y^{\mathrm{tr}}, X^{\mathrm{tr}}))$

Classification of simple ∗-algebras

Theorem (Albert, Jacobson)

A finite simple ∗-algebra is ∗-isomorphic to one of the following types, for some n and q:

1. *Orthogonal:* $\mathrm{O}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto DX^{\mathrm{tr}}D^{-1})$
2. *Symplectic:* $\mathrm{S}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto JX^{\mathrm{tr}}J^{-1})$
3. *Unitary:* $\mathrm{U}(n, \mathbb{F}_{q^2}) = (M_n(\mathbb{F}_{q^2}), X \mapsto \overline{X}^{\mathrm{tr}})$
4. *Exchange:* $\mathrm{X}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q) \oplus M_n(\mathbb{F}_q), (X, Y) \mapsto (Y^{\mathrm{tr}}, X^{\mathrm{tr}}))$

Classification of simple ∗-algebras

Theorem (Albert, Jacobson)

A finite simple ∗-algebra is ∗-isomorphic to one of the following types, for some n and q:

1. *Orthogonal:* $\mathrm{O}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto DX^{\mathrm{tr}}D^{-1})$
2. *Symplectic:* $\mathrm{S}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto JX^{\mathrm{tr}}J^{-1})$
3. *Unitary:* $\mathrm{U}(n, \mathbb{F}_{q^2}) = (M_n(\mathbb{F}_{q^2}), X \mapsto \overline{X}^{\mathrm{tr}})$
4. *Exchange:* $\mathrm{X}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q) \oplus M_n(\mathbb{F}_q), (X, Y) \mapsto (Y^{\mathrm{tr}}, X^{\mathrm{tr}}))$

Classification of simple ∗-algebras

Theorem (Albert, Jacobson)

A finite simple ∗-algebra is ∗-isomorphic to one of the following types, for some n and q:

1. *Orthogonal:* $\mathrm{O}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto DX^{\mathrm{tr}}D^{-1})$
2. *Symplectic:* $\mathrm{S}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto JX^{\mathrm{tr}}J^{-1})$
3. *Unitary:* $\mathrm{U}(n, \mathbb{F}_{q^2}) = (M_n(\mathbb{F}_{q^2}), X \mapsto \overline{X}^{\mathrm{tr}})$
4. *Exchange:* $\mathrm{X}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q) \oplus M_n(\mathbb{F}_q), (X, Y) \mapsto (Y^{\mathrm{tr}}, X^{\mathrm{tr}}))$

Classification of simple ∗-algebras

Theorem (Albert, Jacobson)

A finite simple ∗-algebra is ∗-isomorphic to one of the following types, for some n and q:

1. *Orthogonal:* $\mathrm{O}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto DX^{\mathrm{tr}}D^{-1})$
2. *Symplectic:* $\mathrm{S}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q), X \mapsto JX^{\mathrm{tr}}J^{-1})$
3. *Unitary:* $\mathrm{U}(n, \mathbb{F}_{q^2}) = (M_n(\mathbb{F}_{q^2}), X \mapsto \overline{X}^{\mathrm{tr}})$
4. *Exchange:* $\mathrm{X}(n, \mathbb{F}_q) = (M_n(\mathbb{F}_q) \oplus M_n(\mathbb{F}_q), (X, Y) \mapsto (Y^{\mathrm{tr}}, X^{\mathrm{tr}}))$

Determining the structure of ∗-algebras

Theorem (B. & Wilson, 2009)

There is a Las Vegas algorithm which, given any ∗-algebra $A = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$, *finds each of the following:*

(i) $J(A)$, *the Jacobson radical of* A.

(ii) A *∗-isomorphism* $\rho\colon R/J(A) \to T_1 \oplus \ldots \oplus T_m$, *where* T_j *is a naturally represented simple ∗-algebra.*

(iii) *For* $1 \leqslant j \leqslant m$, *a constructive ∗-isomorphism from* T_j *to a simple ∗-algebra described in the classification.*

Its complexity is $O(|X|\{d^3 + d^2 \log q + d \cdot \mathrm{Star}(A, *)\} + \xi)$.

∗-invariant complements

Theorem (Taft, 1957)

If $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$ *is a* $*$*-algebra and* $|\mathbb{F}_q|$ *is odd, then there is a* $*$*-subalgebra* $S \subseteq A$ *such that* $A = J(A) \oplus S$.

We refer to the decomposition of A as a **Taft decomposition**.

Theorem (B. & Wilson, 2009)

There is a Las Vegas algorithm which, for any given $*$*-algebra* $A = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$ *(*q *odd), returns a Taft decomposition of* A. *Its complexity is* $O(|X|\{d^3 + d^2 \log q + \mathrm{Star}(A, *)\} + \xi)$.

∗-invariant complements

Theorem (Taft, 1957)

If $A \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$ *is a* $*$*-algebra and* $|\mathbb{F}_q|$ *is odd, then there is a* $*$*-subalgebra* $S \subseteq A$ *such that* $A = J(A) \oplus S$.

We refer to the decomposition of A as a **Taft decomposition**.

Theorem (B. & Wilson, 2009)

There is a Las Vegas algorithm which, for any given $*$*-algebra* $A = \mathrm{Env}(X) \leqslant \mathrm{End}_{\mathbb{F}_q}(V)$ *(*q *odd), returns a Taft decomposition of* A. *Its complexity is* $O(|X|\{d^3 + d^2 \log q + \mathrm{Star}(A, *)\} + \xi)$.

p-groups and the *stabilizer problem*

One approach to computing automorphism groups of p-groups gives rise to the following problem:

- Given a space Φ of alternating forms on an $\mathbb{F}_q$-space V.
- Find the subgroup G_Φ of $\mathrm{GL}(V)$ stabilizing Φ, namely

$$G_\Phi = \{g \in \mathrm{GL}(V) \colon \varphi^g \in \Phi \text{ for all } \varphi \in \Phi\}$$

Isometries and the *centralizer problem*

One can interpret the space of forms Φ instead as a bilinear map $b\colon V \times V \to W$, in which case G_Φ is precisely

$$\Psi\mathrm{Isom}(b) = \{(g,h) \in \mathrm{GL}(V) \times \mathrm{GL}(W)\colon b(ug,vg) = b(u,v)h\}$$

the group of **pseudo-isometries** of the bilinear map b. This contains the **isometry group** of b as a normal subgroup:

$$\mathrm{Isom}(b) = \{g \in \mathrm{GL}(V)\colon b(ug,vg) = b(u,v)\}$$

Translating back to Φ, the isometry group of b corresponds to the group that **centralizes** the space of forms.

Isometries and the *centralizer problem*

One can interpret the space of forms Φ instead as a bilinear map $b\colon V \times V \to W$, in which case G_Φ is precisely

$$\Psi\mathrm{Isom}(b) = \{(g,h) \in \mathrm{GL}(V) \times \mathrm{GL}(W)\colon b(ug,vg) = b(u,v)h\}$$

the group of **pseudo-isometries** of the bilinear map b. This contains the **isometry group** of b as a normal subgroup:

$$\mathrm{Isom}(b) = \{g \in \mathrm{GL}(V)\colon b(ug,vg) = b(u,v)\}$$

Translating back to Φ, the isometry group of b corresponds to the group that **centralizes** the space of forms.

Isometries and the *centralizer problem*

One can interpret the space of forms Φ instead as a bilinear map $b\colon V \times V \to W$, in which case G_Φ is precisely

$$\Psi\mathrm{Isom}(b) = \{(g,h) \in \mathrm{GL}(V) \times \mathrm{GL}(W)\colon b(ug,vg) = b(u,v)h\}$$

the group of **pseudo-isometries** of the bilinear map b. This contains the **isometry group** of b as a normal subgroup:

$$\mathrm{Isom}(b) = \{g \in \mathrm{GL}(V)\colon b(ug,vg) = b(u,v)\}$$

Translating back to Φ, the isometry group of b corresponds to the group that **centralizes** the space of forms.

Constructing isometry groups

Theorem (B. & Wilson, 2009)

There is a polynomial-time algorithm which, given any Hermitian map $b\colon V \times V \to W$*, where* V*,* W *are* $\mathbb{F}_q$*-spaces* ***and*** $|\mathbb{F}_q|$ ***is odd****, returns generators for* $\mathrm{Isom}(b)$*.*

Remarks:

1. The algorithm also describes the structure of $\mathrm{Isom}(b)$ explicitly; in particular it easily determines the order of this group.
2. The complexity of the algorithm is $O(d^6)$.

Constructing isometry groups

Theorem (B. & Wilson, 2009)

There is a polynomial-time algorithm which, given any Hermitian map $b\colon V \times V \to W$, *where* V, W *are* $\mathbb{F}_q$*-spaces* ***and*** $|\mathbb{F}_q|$ ***is odd***, *returns generators for* $\mathrm{Isom}(b)$.

Remarks:

1. The algorithm also describes the structure of $\mathrm{Isom}(b)$ explicitly; in particular it easily determines the order of this group.
2. The complexity of the algorithm is $O(d^6)$.

Constructing isometry groups

Theorem (B. & Wilson, 2009)

There is a polynomial-time algorithm which, given any Hermitian map $b\colon V \times V \to W$, *where* V, W *are* $\mathbb{F}_q$*-spaces* ***and*** $|\mathbb{F}_q|$ ***is odd***, *returns generators for* $\mathrm{Isom}(b)$.

Remarks:

1. The algorithm also describes the structure of $\mathrm{Isom}(b)$ explicitly; in particular it easily determines the order of this group.
2. The complexity of the algorithm is $O(d^6)$.

Constructing isometry groups

Theorem (B. & Wilson, 2009)

There is a polynomial-time algorithm which, given any Hermitian map $b\colon V \times V \to W$, *where* V, W *are* $\mathbb{F}_q$*-spaces* ***and*** $|\mathbb{F}_q|$ ***is odd***, *returns generators for* $\mathrm{Isom}(b)$.

Remarks:

1. The algorithm also describes the structure of $\mathrm{Isom}(b)$ explicitly; in particular it easily determines the order of this group.
2. The complexity of the algorithm is $O(d^6)$.

From adjoints to isometries

If A is any $*$-algebra, we define

$$A^{\#} = \{x \in A : x^*x = xx^* = 1\}$$

the **group of unitary elements** of A.

$$\begin{aligned}
\mathrm{Isom}(b) &= \{g \in \mathrm{GL}(V) : b(ug, vg) = b(u, v) \text{ for all } u, v \in V\} \\
&= \{g \in \mathrm{GL}(V) : (g, g^{-1}) \in \mathrm{Adj}(b)\} \\
&= \{g \in \mathrm{Adj}(b) : g^*g = gg^* = 1\} \\
&= \mathrm{Adj}(b)^{\#}
\end{aligned}$$

From adjoints to isometries

If A is any $*$-algebra, we define

$$A^{\#} = \{x \in A : x^*x = xx^* = 1\}$$

the **group of unitary elements** of A.

$$\begin{aligned}
\mathrm{Isom}(b) &= \{g \in \mathrm{GL}(V) : b(ug, vg) = b(u, v) \text{ for all } u, v \in V\} \\
&= \{g \in \mathrm{GL}(V) : (g, g^{-1}) \in \mathrm{Adj}(b)\} \\
&= \{g \in \mathrm{Adj}(b) : g^*g = gg^* = 1\} \\
&= \mathrm{Adj}(b)^{\#}
\end{aligned}$$

Using the structure of $\mathrm{Adj}(b)$

Since $\mathrm{char}(\mathbb{F}_q) > 2$ we can decompose $\mathrm{Adj}(b)$ as follows:

$$\mathrm{Adj}(b) = J(\mathrm{Adj}(b)) \oplus (T_1 \oplus \ldots \oplus T_m)$$

where T_i is a simple $*$-algebra defined naturally on a $\mathbb{K}_i$-space W_i.

This decomposition gives rise to one for $\mathrm{Adj}(b)^{\#}$:

$$\mathrm{Adj}(b)^{\#} = O_p(\mathrm{Adj}(b)^{\#}) \rtimes (T_1^{\#} \times \ldots \times T_m^{\#})$$

where $T_i^{\#}$ is a classical group defined naturally on W_i.

Using the structure of Adj(b)

Since char($\mathbb{F}_q$) > 2 we can decompose Adj(b) as follows:

$$\mathrm{Adj}(b) = J(\mathrm{Adj}(b)) \oplus (T_1 \oplus \ldots \oplus T_m)$$

where T_i is a simple $*$-algebra defined naturally on a $\mathbb{K}_i$-space W_i. This decomposition gives rise to one for Adj(b)$^{\#}$:

$$\mathrm{Adj}(b)^{\#} = O_p(\mathrm{Adj}(b)^{\#}) \rtimes (T_1^{\#} \times \ldots \times T_m^{\#})$$

where $T_i^{\#}$ is a classical group defined naturally on W_i.

Constructing the unipotent radical $O_p(\mathrm{Adj}(b)^{\#})$

It is customary to use exponentiation to construct unipotent elements in an algebraic group from nilpotent elements of its associated Lie algebra. This approach founders, however, when the nilpotence class exceeds the characteristic.

Instead, we use the following map, which only requires division by 2:

$$x \mapsto x + \sqrt{1+x^2} = 1 + x - 2\sum_{n=1}^{\infty} C(n-1)\left(-\frac{1}{4}\right)^2 x^n$$

[$C(j)$ denotes the j^{th} Catalan number]

This is another place our approach falls down in characteristic 2!

Constructing the unipotent radical $O_p(\mathrm{Adj}(b)^{\#})$

It is customary to use exponentiation to construct unipotent elements in an algebraic group from nilpotent elements of its associated Lie algebra. This approach founders, however, when the nilpotence class exceeds the characteristic.

Instead, we use the following map, which only requires division by 2:

$$x \mapsto x + \sqrt{1+x^2} = 1 + x - 2\sum_{n=1}^{\infty} C(n-1)\left(-\frac{1}{4}\right)^2 x^n$$

[$C(j)$ denotes the j^{th} Catalan number]

This is another place our approach falls down in characteristic 2!

Constructing the unipotent radical $O_p(\mathrm{Adj}(b)^{\#})$

It is customary to use exponentiation to construct unipotent elements in an algebraic group from nilpotent elements of its associated Lie algebra. This approach founders, however, when the nilpotence class exceeds the characteristic.

Instead, we use the following map, which only requires division by 2:

$$x \mapsto x + \sqrt{1+x^2} = 1 + x - 2\sum_{n=1}^{\infty} C(n-1)\left(-\frac{1}{4}\right)^2 x^n$$

[$C(j)$ denotes the j^{th} Catalan number]

This is another place our approach falls down in characteristic 2!

Concluding remarks

1. Isometry groups of bilinear maps are intersections of classical groups. Our algorithm can be adapted to provide a polynomial-time algorithm to construct the intersection of a given collection of classical groups defined naturally on a vector space of odd order.
2. We have implemented the various algorithms in MAGMA.
3. Can these methods be used to investigate problems concerning group algebras?
4. Constructing and describing the structure of $\Psi\mathrm{Isom}(b)$ seems to be a harder problem. However, using the structure of $\mathrm{Adj}(b)$, we can construct generators for the group

$$\mathrm{N}^{*}_{\mathrm{GL}(V)}(\mathrm{Adj}(b)) = \{x \in \mathrm{GL}(V) : f^x \in \mathrm{Adj}(b) \text{ and } (f^x)^* = (f^*)^x\}$$

which contains $\Psi\mathrm{Isom}(b)$ as a normal subgroup.

Concluding remarks

1. Isometry groups of bilinear maps are intersections of classical groups. Our algorithm can be adapted to provide a polynomial-time algorithm to construct the intersection of a given collection of classical groups defined naturally on a vector space of odd order.
2. We have implemented the various algorithms in MAGMA.
3. Can these methods be used to investigate problems concerning group algebras?
4. Constructing and describing the structure of $\Psi\mathrm{Isom}(b)$ seems to be a harder problem. However, using the structure of $\mathrm{Adj}(b)$, we can construct generators for the group

$$\mathrm{N}^{*}_{\mathrm{GL}(V)}(\mathrm{Adj}(b)) = \{x \in \mathrm{GL}(V) : f^x \in \mathrm{Adj}(b) \text{ and } (f^x)^* = (f^*)^x\}$$

which contains $\Psi\mathrm{Isom}(b)$ as a normal subgroup.

Concluding remarks

1. Isometry groups of bilinear maps are intersections of classical groups. Our algorithm can be adapted to provide a polynomial-time algorithm to construct the intersection of a given collection of classical groups defined naturally on a vector space of odd order.
2. We have implemented the various algorithms in MAGMA.
3. Can these methods be used to investigate problems concerning group algebras?
4. Constructing and describing the structure of $\Psi\mathrm{Isom}(b)$ seems to be a harder problem. However, using the structure of $\mathrm{Adj}(b)$, we can construct generators for the group

$$\mathrm{N}^{*}_{\mathrm{GL}(V)}(\mathrm{Adj}(b)) = \{x \in \mathrm{GL}(V) : f^x \in \mathrm{Adj}(b) \text{ and } (f^x)^* = (f^*)^x\}$$

which contains $\Psi\mathrm{Isom}(b)$ as a normal subgroup.

Concluding remarks

1. Isometry groups of bilinear maps are intersections of classical groups. Our algorithm can be adapted to provide a polynomial-time algorithm to construct the intersection of a given collection of classical groups defined naturally on a vector space of odd order.
2. We have implemented the various algorithms in MAGMA.
3. Can these methods be used to investigate problems concerning group algebras?
4. Constructing and describing the structure of $\Psi\mathrm{Isom}(b)$ seems to be a harder problem. However, using the structure of $\mathrm{Adj}(b)$, we can construct generators for the group

$$\mathrm{N}^{*}_{\mathrm{GL}(V)}(\mathrm{Adj}(b)) = \{x \in \mathrm{GL}(V) : f^x \in \mathrm{Adj}(b) \text{ and } (f^x)^* = (f^*)^x\}$$

which contains $\Psi\mathrm{Isom}(b)$ as a normal subgroup.

Concluding remarks

1. Isometry groups of bilinear maps are intersections of classical groups. Our algorithm can be adapted to provide a polynomial-time algorithm to construct the intersection of a given collection of classical groups defined naturally on a vector space of odd order.
2. We have implemented the various algorithms in MAGMA.
3. Can these methods be used to investigate problems concerning group algebras?
4. Constructing and describing the structure of $\Psi\mathrm{Isom}(b)$ seems to be a harder problem. However, using the structure of $\mathrm{Adj}(b)$, we can construct generators for the group

$$\mathrm{N}^{*}_{\mathrm{GL}(V)}(\mathrm{Adj}(b)) = \{x \in \mathrm{GL}(V) : f^x \in \mathrm{Adj}(b) \text{ and } (f^x)^* = (f^*)^x\}$$

which contains $\Psi\mathrm{Isom}(b)$ as a normal subgroup.