REPRESENTATIONS OF ALGEBRAS: SOME COMPUTATIONAL ASPECTS

A Thesis
Submitted to
the Temple University Graduate Board

---

in Partial Fulfillment
of the Requirements for the Degree of
MASTER OF SCIENCE

---

by
Kyle Rhoads
August, 2019

Thesis Approvals:
Dr. Edward Letzter, Thesis Advisor, Department of Mathematics
Dr. Martin Lorenz, Department of Mathematics
Dr. Vasily Dolgushev, Department of Mathematics
Dr. Peter Riseborough, Temple University, Department of Physics

ABSTRACT

REPRESENTATIONS OF ALGEBRAS: SOME COMPUTATIONAL ASPECTS

Kyle Rhoads

MASTER OF SCIENCE

Temple University, August, 2019

Professor Edward S. Letzter, Chair

Let $n \in \mathbb{Z}_{\geq 1}$ and R be a finitely presented k-algebra over a computable field k. We describe algorithms for computably deciding representation-theoretic properties of R, following work by Letzter. Among these algorithms is an algorithm deciding whether an n-dimensional irreducible representation exists. We also provide a package, `finitely-presented-algebra`, which implements these algorithms into the computer algebra system `SAGE`.

ACKNOWLEDGEMENTS

First and foremost, I must thank my advisor, Edward Letzter, who has patiently guided and encouraged my growth during my time at Temple University. My earnest exploration of several areas of interest, which culminated in this thesis, would not have been possible without his invaluable support. For this I am forever grateful.

Second, I would like to thank my committee members Vasily Dolgushev and Martin Lorenz, who were key in strengthening my passion for algebra. Their feedback and participation was essential to the preparation and defense of my thesis. I must also express my sincere appreciation for Peter Riseborough, as the external reader for my examining committee.

I would be remiss if I did not extend heartfelt gratitude to my friends and colleagues at Temple, especially Brandi Henry and Rose Kaplan-Kelly. Temple's mathematics community has fostered growth and passion within me in a way that no other can. This appreciation extends to each professor I had the privilege of learning from during my time at Temple.

I would like to thank my family for their never-ending support. I would never have been able to begin this journey without their help. Finally, I would like to thank my partner, Chelsea Cole, who inspires me everyday to be the best version of myself.

TABLE OF CONTENTS

ABSTRACT **iv**

ACKNOWLEDGEMENT **v**

1 INTRODUCTION **1**

2 PRELIMINARIES **4**
2.1 Notation 4
2.2 Representations 6

3 COMPUTATIONS IN COMMUTATIVE ALGEBRA **10**
3.1 Factoring Algorithms 11
3.2 Groebner Bases 12
3.3 Division and Buchberger's Algorithm 14
3.4 The Ideal and Radical Membership Algorithms 16

4 MATRIX ALGEBRAS AND LENGTH **19**
4.1 Generating Sets and Paz's Conjecture 19
4.2 Instances of Linear Bounds 22

5 DECIDABILITY FOR REPRESENTATIONS **25**
5.1 General Representations 25
5.2 Irreducibility 27
5.3 Semisimplicity 33

6 THE SOFTWARE **34**
6.1 The Basics 35
6.2 Algorithm for Representations 37
6.2.1 Tests for Representation 37
6.2.2 Testing if a Representation is Irreducible 39
6.2.3 Testing Existence of Irreducible Representations 40

6.3 Performance 42
6.4 Examples 44
6.4.1 First Weyl Algebra 44
6.4.2 Projective Special Linear Group 45
6.4.3 Off-diagonal Generating Set 47

REFERENCES **49**

CHAPTER 1

INTRODUCTION

One of the best ways to understand some algebraic structures is through their representations. This is true in the case of associative algebras. One useful class of associative algebras is the class of finitely presented algebras, which are free algebras modulo finitely generated ideals. This endows the algebra with a presentation structure, which is desirable as it allows these algebras to be approached algorithmically, in principle. Interesting examples include quantum algebras, quaternion algebras, the Weyl algebras, and group algebras for finitely presented groups.

This leads to a natural question: given a finitely presented algebra R, what representation-theoretic properties of R can be decided computationally? in [8, 9, 10], Letzter provides algorithms that can, in principle, decide representation-theoretic properties of finitely presented algebras, when working over a computable field. Among these include determining whether an irreducible representation of R exists in a fixed finite dimension, and determining if a given map is an irreducible representation.

The purpose of this paper is twofold. First, we lay ground work from computational commutative algebra and linear algebra, and then present the representation-theoretic results proved by Letzter. Second, we present our implementations of several of these algorithms into the computer algebra system `SAGE`. The organization of the paper is outlined below.

Beginning in Chapter 2, we present the notation and definitions which will be used throughout. This includes defining computability, defining finitely presented algebras, and discussing their representations.

In Chapter 3, we outline main definitions and theorems from computational commutative algebra to be used later. This primarily includes discussion of Groebner bases, in order to present the ideal membership algorithm and radical membership algorithm.

In Chapter 4, we briefly discuss generating sets for matrix algebras, and Paz's conjecture for lengths of matrix algebras. We then give a preview of several results made towards Paz's conjecture.

In Chapter 5, we present the decidable representation-theoretic properties of finitely presented algebras, as proved by Letzter. This includes decisions towards general representation properties in finite dimensions, as well as decisions towards irreducibile and semisimple representations.

Lastly, in Chapter 6, we provide our package `finitely_presented_algebra` in the computer algebra system `SAGE`. We discuss usage of the package, and provide examples of results obtained for representations of specific algebras using the package.

CHAPTER 2

PRELIMINARIES

In this chapter we will present the notation and conventions that will be used throughout, define finitely presented algebras, and then discuss representations and their properties.

2.1 Notation

To begin, when we refer to a process or object as computable, we mean that it can be implemented, in principle, into an idealized computer algebra system without loss of precision. When we refer to an algorithm or test, we will always mean a computable process. Often we will ask that a field be computable. Examples of computable fields include $\mathbb{Q}$, and finite fields $\mathbb{F}_q$. Nonexamples include $\mathbb{R}$ and $\mathbb{C}$.

A question is decidable if there exists a computable process that guarantees an answer to the question in finite time. A property P of an object X is decidable if the question “does X have property P” is decidable.

Let k be a field of arbitrary characteristic. All k-algebras will be assumed to be unital and associative, unless otherwise stated. Let $k\{X_1, \ldots, X_m\}$ denote the free k-algebra on m generators. For $f_1, \ldots f_t \in k\{X_1, \ldots, X_m\}$, we will use $\langle f_1, \ldots, f_t \rangle$ to denote the ideal of $k\{X_1, \ldots, X_m\}$ generated by $f_1, \ldots, f_t$.

Definition 2.1. Let $f_1, \ldots, f_t \in k\{X_1, \ldots, X_m\}$. Set

$$R = \frac{k\{X_1, \ldots, X_m\}}{\langle f_1, \ldots, f_t \rangle}.$$

Here we call R a **finitely presented k-algebra**. Further, we denote R by $k\langle X_1, \ldots, X_m \mid f_1, \ldots, f_t \rangle$, and call this a **presentation of** R.

In a slight abuse of notation, $X_1, \ldots, X_m$ will also denote their images in R, so that $f_1 = \cdots = f_t = 0$ in R without a change in notation.

Example 2.2. In possibly the simplest example, the polynomial ring over k in two variables $k[x, y]$ can be formed by quotienting the free algebra $k\{x, y\}$ by the ideal $\langle xy - yx \rangle$. Thus, $k[x, y] = k\langle x, y \mid xy - yx \rangle$. This of course can be done for $k[x_1, \ldots, x_n]$ for any $n \in \mathbb{Z}_{\geq 1}$.

Example 2.3. We define the first Weyl algebra over k as

$$A_1(k) = \frac{k\{X, Y\}}{\langle YX - XY - 1 \rangle} = k\langle X, Y \mid YX - XY = 1 \rangle.$$

The first Weyl algebra is the first in an infinite family of algebras that are used to study differential operators, as well as quantum mechanics. Representations of this algebra will be discussed in Example 2.7.

Example 2.4. Let $0 \neq q \in k$. We call the following algebra the quantum plane:

$$k_q[X, Y] = \frac{k\{X, Y\}}{\langle YX - qXY \rangle} = k\langle X, Y \mid YX = qXY \rangle.$$

This gives rise to a class of finitely presented k-algebras, dependent on $q \in k$. Often, we will want to note whether q is a root of unity.

Example 2.5. Let $G = \langle g_1, \ldots, g_m \mid r_1, \ldots, r_s \rangle$ be a finitely presented group. Then $k\langle g_1, \ldots, g_m, \overline{g_1}, \ldots, \overline{g_m} \mid r_1 - 1, \ldots, r_s - 1, g_1\overline{g_1} - 1, \ldots, g_m\overline{g_m} - 1 \rangle$ is equivalent to the group algebra $k[G]$. Here, $\overline{g_i}$ denotes the inverse of g_i. Thus finitely presented groups give rise to finitely presented algebras.

The representation theory of the group G over k coincides with the representation theory of $k[G]$, allowing us to gain information about G by examining $k[G]$. An example of such is discussed in 6.4.2.

2.2 Representations

Fix n as a positive integer, and let $M_n(\overline{k})$ denote the $\overline{k}$-algebra of $n \times n$ matrices over $\overline{k}$, where $\overline{k}$ is the algebraic closure of k. Note that $M_n(\overline{k})$ is a k-algebra, as well.

Definition 2.6. Let R be a k-algebra.

1. If $\rho : R \to M_n(\overline{k})$ is a k-algebra homomorphism, we say that ρ is a n**-dimensional representation of** R.

2. Two representations $\rho_1, \rho_2 : R \to M_n(\overline{k})$ are **equivalent** if there exists a matrix $Q \in \mathrm{GL}_n(\overline{k})$ such that $\rho_1(r) = Q\rho_2(r)Q^{-1}$ for all $r \in R$.

3. If $\rho(R)$ $\overline{k}$-linearly spans $M_n(\overline{k})$, we say that ρ is **irreducible**. That is, $\overline{k}\rho(R) = M_n(\overline{k})$.

4. We call ρ **semisimple** if ρ is equivalent to a representation of the form

$$r \mapsto \begin{bmatrix} \rho_1(r) & & & \\ & \rho_2(r) & & \\ & & \ddots & \\ & & & \rho_r(r) \end{bmatrix},$$

   where each $\rho_i : R \to M_{n_i}(\overline{k})$ is an irreducible n_i-dimensional representation, for suitable choices of $n_1, \ldots, n_r \in \mathbb{Z}_{\geq 1}$.

It is worth noting that this not the standard definition of a representation. In the standard definition, the codomain of ρ is $M_n(k)$, instead of $M_n(\overline{k})$. Our definition, as it is stated in [8, 9, 10], appears in the foundational work [1] by Artin. This definition is useful as it allows us to find irreducible or semisimple representations over $\overline{k}$ by working over its subfield k, for which it is often easier to directly obtain results.

In some cases, finite dimensional representations can be well understood.

Example 2.7. Let $A_1(k) = k\langle X, Y \mid XY - YX = 1\rangle$, the first Weyl algebra. Assuming k is a field of characteristic zero, $A_1(k)$ has no finite dimensional representations.

To see this, fix a positive integer n and suppose $\rho : A_1(k) \to M_n(\overline{k})$ is a representation. Let $\boldsymbol{x} = \rho(X)$ and $\boldsymbol{y} = \rho(Y)$, and note that both $\boldsymbol{x}$ and $\boldsymbol{y}$ must be nonzero. We know that for any two matrices $A, B \in M_n(\overline{k})$, $\text{trace}(AB) = \text{trace}(BA)$. Therefore $\text{trace}(\boldsymbol{xy} - \boldsymbol{yx}) = 0$. However, since ρ is a homomorphism, and $\rho(1_k) = I_n$, the identity of $M_n(\overline{k})$, this implies that $0 = \text{trace}(\boldsymbol{xy} - \boldsymbol{yx}) = \text{trace}(I_n) = n$, a contradiction. Thus no finite dimensional representations of $A_1(k)$ exist.

Example 2.8. Recall our definition $k_q[X, Y] = k\langle X, Y \mid YX = qXY\rangle$ for nonzero $q \in k$, as in Example 2.4. We can similarly define

$$k_q[X^{\pm 1}, Y^{\pm 1}] = k\langle X, Y, U, V \mid YX = qXY, UX = XU = VY = YV = 1\rangle,$$

which we call the quantum torus. Note that $U = X^{-1}$ and $V = Y^{-1}$. We have that $k_q[X^{\pm 1}, Y^{\pm 1}]$ is a simple k-algebra, meaning it has no proper nontrivial two-sided ideals, if and only if q is not a root of unity in k; see [5] for one such proof of this. Additionally, $k_q[X^{\pm 1}, Y^{\pm 1}]$ is infinite dimensional as a k-vector space. These two results can be used to show that $k_q[X^{\pm 1}, Y^{\pm 1}]$ has no finite dimensional irreducible representations, when q is not a root of unity in k.

In situations where one has not, or cannot, determine the existence of irreducible representations through more general means, computational approaches are worth exploring. Results obtained computationally can be used to influence the direction of theoretical research, as well as directly study specific instances of interesting algebraic objects. The following chapters will outline material we will use to approach representations from a computational standpoint.

CHAPTER 3

COMPUTATIONS IN COMMUTATIVE ALGEBRA

This chapter will present foundational results in computational commutative algebra, which we will then later use when discussing decidable properties of finitely presented algebras. Most of the material discussed will be presented without proof; for a more complete detailing of the subject complete with proofs, we recommend [3].

Computational commutative algebra has proved to be highly successful in many diverse mathematical settings. It is thus desirable to translate problems working in a noncommutative algebra into the language of commutative algebra, in order to gain access to symbolic computation. Particularly, for a finitely presented algebra $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t \rangle$, we often want to form ideals of a commutative polynomial ring based on the noncommutative polynomials $f_1, \ldots, f_t$. We then use these ideals to gain information about R.

3.1 Factoring Algorithms

Let k be an arbitrary field, and let $k[x]$ denote the univariate polynomial ring over k. Recall that $k[x]$ is a unique factorization domain, and so every polynomial in $k[x]$ can represented uniquely as a scaled product of irreducible polynomials in $k[x]$.

Definition 3.1. By a **factoring algorithm for** $k[x]$, we mean an algorithm which takes nonzero $f \in k[x]$ as input, and gives as output $0 \neq a \in k$ and irreducible polynomials $g_1, \ldots, g_s \in k[x]$, such that $f = a \cdot g_1 \cdots g_s$. That is, an algorithm which factors f in $k[x]$.

Not every polynomial ring over a field has such a factoring algorithm. In particular, if k is not computable, then $k[x]$ cannot have a factoring algorithm. However, if we have do have such a factoring algorithm for $k[x]$, this algorithm can be used to solve polynomial equations over larger multivariate polynomial rings $k[x_1, \ldots, x_n]$. Some of the algorithms outlined in Chapter 5 will require such an algorithm in order to be computable.

3.2 Groebner Bases

We again assume that k is an arbitrary field and n is a positive integer, and let $k[x_1, \ldots, x_n]$ denote the polynomial ring in n commutating variables. For our purposes, taking $f_1, \ldots, f_s \in k[x_1, \ldots, x_n]$, we will use $\langle f_1, \ldots, f_s \rangle$ to denote the ideal generated by $f_1, \ldots, f_s$, and $(f_1, \ldots, f_s)$ to denote the ordered tuple containing $f_1, \ldots, f_s$.

Definition 3.2. A **monomial ordering on** $k[x_1, \ldots, x_n]$ is a total ordering $\leq$ on the set of monomials of $k[x_1, \ldots, x_n]$, such that

1. if w, y, z are monomials such that $w \leq y$, then $wz \leq yz$.
2. $\leq$ is a well ordering.

Example 3.3. The following is a classic example, called the lexicographic ordering: for two monomials $x_1^{s_1} \cdots x_n^{s_n}, x_1^{t_1} \cdots x_n^{t_n} \in k[x_1, \ldots, x_n]$ with $s_i, t_i \geq 0$ for all i, we say that $x_1^{s_1} \cdots x_n^{s_n} \leq x_1^{t_1} \cdots x_n^{t_n}$ if

1. $\min\{i \mid s_i > 0\} \leq \min\{i \mid t_i > 0\}$,
2. or if $j = \min\{i \mid s_i > 0\} = \min\{i \mid t_i > 0\}$ then $s_j \leq t_j$.

One can always endow $k[x_1, \ldots, x_n]$ with a monomial ordering for arbitrary k and n, for example by giving it the lexicographic ordering. Moving forward, we will always assume that $k[x_1, \ldots, x_n]$ is given a fixed monomial ordering.

Definition 3.4. Let $f \in k[x_1, \ldots, x_n]$.

- Let $\mathrm{LM}(f)$ denote the largest monomial appearing in f, with respect to the monomial ordering of $k[x_1, \ldots, x_n]$, which we call the **leading monomial of** f.
- Let $\mathrm{LC}(f)$ denote the coefficient attached to $\mathrm{LM}(f)$, the **leading coefficient of** f.
- Let $\mathrm{LT}(f) = \mathrm{LC}(f) \cdot \mathrm{LM}(f)$, the **leading term of** f.

Definition 3.5. Let I be a nonzero ideal of $k[x_1, \ldots, x_n]$. Let

$$\mathrm{LT}(I) = \{ax_1^{t_1} \cdots x_n^{t_n} \mid \text{ there exists } f \in I \text{ with } \mathrm{LT}(f) = ax_1^{t_1} \cdots x_n^{t_n}\}.$$

We call $\mathrm{LT}(I)$ the **leading terms of** I.

Note that $\langle \mathrm{LT}(I) \rangle$ is an ideal of $k[x_1, \ldots, x_n]$, and that $\langle \mathrm{LT}(I) \rangle$ is not necessarily equal to I.

Definition 3.6. A subset $G = \{g_1, \ldots, g_s\}$ of an ideal I is said to be a **Groebner basis** if $\langle \mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_s) \rangle = \langle \mathrm{LT}(I) \rangle$.

Proposition 3.7. *Let I be a nonzero ideal of $k[x_1, \ldots, x_n]$. Then I has a Groebner basis, and any Groebner basis of I is a generating set of I.*

This establishes the existence of Groebner bases for an arbitrary ideal of a commutative polynomial ring. However, for computability we must be able to generate a Groebner basis for an ideal in an algorithmic manner.

3.3 Division and Buchberger's Algorithm

Proposition 3.8. *Let* $F = (f_1, \ldots, f_s) \subseteq k[x_1, \ldots, x_n]$ *be an ordered tuple. Let* $f \in k[x_1, \ldots, x_n]$*. Then there exist* $a_1, \ldots, a_s, r \in k[x_1, \ldots, x_n]$ *such that*

$$f = a_1 f_1 + \cdots + a_s f_s + r,$$

and either $r = 0$*, or no term of* r *is divisible by* $\mathrm{LT}(f_i)$ *for every* i*.*

Definition 3.9. We call r as above a **remainder of** f **from division by** F, and we denote r by $\overline{f}^F$.

In particular, there exists an algorithm to produce such $a_1, \ldots, a_s, r$ which is referred to as the division algorithm in $k[x_1, \ldots, x_n]$. For our purposes we will not prove the validity of the division algorithm, but instead accept its existence.

Another important result is Hilbert's basis theorem. Recall that a ring R, not necessarily commutative, is called left Noetherian if every ascending chain of left ideals of R terminates after finitely many steps, or equivalently if every left ideal of R has a finite generating set. We similarly define right Noetherian for right ideals. Then R is called Noetherian if R is both left Noetherian and right Noetherian. Note that all three definitions coincide when R is commutative, since in this case left ideals are rights ideals, and vice versa.

Theorem 3.10. *(Hilbert's basis theorem) Let* R *be a ring. If* R *is Noetherian, then* $R[x]$ *is Noetherian.*

Clearly any field is Noetherian, which immediately gives us that $k[x_1, \ldots, x_n]$ is Noetherian. Moving forward we will assume that we always are given a finite generating set $f_1, \ldots, f_s$ for an ideal I. That is, $I = \langle f_1, \ldots, f_s \rangle$. This leads us to Buchberger's algorithm.

Theorem 3.11. *(Buchberger's algorithm) Let $I = \langle f_1, \ldots, f_s \rangle$ be a nonzero ideal of $k[x_1, \ldots, x_n]$. The following algorithm produces a Groebner basis G of I:*

input : $(f_1, \ldots, f_s)$, **require** $f_1, \ldots, f_s$ **in** $k[x_1, \ldots, x_n]$

begin algorithm

$G := (f_1, \ldots, f_s)$

$G' := G$

for (p, q) **in** $G' \times G'$, $p \neq q$ **do**

$$S := \frac{\mathrm{lcm}(\mathrm{LM}(p), \mathrm{LM}(q))}{\mathrm{LT}(p)} \cdot p - \frac{\mathrm{lcm}(\mathrm{LM}(p), \mathrm{LM}(q))}{\mathrm{LT}(q)} \cdot q$$

if $\overline{S}^{G'} \neq 0$ **do**

$G := G \cup \left\{ \overline{S}^{G'} \right\}$

if $G = G'$ **do**

return G

end

Above, $\mathrm{lcm}(f, g)$ denotes the least common multiple of f and g. Note that we always have $\langle \mathrm{LT}(G') \rangle \subseteq \langle \mathrm{LT}(G) \rangle$. The key step in showing that this algorithm terminates is that if $G' \neq G$ then $\langle \mathrm{LT}(G') \rangle \subset \langle \mathrm{LT}(G) \rangle$ properly. This is due to the fact that if $G \neq G'$ then $\overline{S}^{G'}$ yields a nonzero remainder r, and so $\mathrm{LT}(r) \notin \langle \mathrm{LT}(G') \rangle$, but by construction $\mathrm{LT}(r) \in \langle \mathrm{LT}(G) \rangle$. Thus we construct an ascending chain of ideals given by $\langle \mathrm{LT}(G') \rangle \subseteq \langle \mathrm{LT}(G) \rangle$, which must eventually terminate since $k[x_1, \ldots, x_n]$ is Noetherian, implying that $G' = G$ after some finite step. Thus the algorithm must terminate.

Be aware that this is not the most efficient version of Buchberger's algorithm. Modern computer algebra systems implement a modified version of the above in order to reduce calculation time; see Section 2.9 of [3] for more elaboration. In general computing the complexity of Buchberger's is not an easy task, but it was shown in [4] that the degrees of elements of G are bounded above by $2((d^2/2) + d)^{2^{n-1}}$, where $d = \max\{\deg(f_1), \ldots, \deg(f_s)\}$.

3.4 The Ideal and Radical Membership Algorithms

We now know that we can compute Groebner bases for a given ideal. Next, we will see how to apply this to determining element containment for ideals.

Proposition 3.12. *Let $f \in k[x_1, \ldots, x_n]$, I be a nonzero ideal of $k[x_1, \ldots, x_n]$, and $G = \{g_1, \ldots, g_s\}$ be a Groebner basis of I. Then there exists a unique $r \in k[x_1, \ldots, x_n]$ such that*

1. *either* $r = 0$, *or no term of* r *is divisible by* $\mathrm{LT}(g_i)$ *for every* i,

2. *there is a* $g \in I$ *such that* $f = g + r$.

Note the similarity to 3.8. What this tells us is that when finding the remainder of f of division by a Groebner basis G, we gain uniqueness, regardless of the order of the elements of G. Recall that this remainder is computable via the division algorithm.

Corollary 3.13. *Let* $G = \{g_1, \ldots, g_s\}$ *be a Groebner basis for an ideal* I *of* $k[x_1, \ldots, x_n]$ *and* $f \in k[x_1, \ldots, x_n]$. *Then determining if* $f \in I$ *is decidable, by the following:* $f \in I$ *if and only if* $\overline{f}^G$ *is zero.*

This gives us our desired result. For I an ideal and $f \in k[x_1, \ldots, x_n]$ we can now algorithmically determine if $f \in I$, by first applying Buchberger's algorithm to gain a Groebner basis G of I, and computing whether $\overline{f}^G = 0$. This process is referred to as the ideal membership algorithm.

Now recall that for an ideal I of $k[x_1, \ldots, x_n]$, the radical of I, denoted $\sqrt{I}$, is $\{f \mid f^k \in I \text{ for some } k\}$. This is also an ideal of $k[x_1, \ldots, x_n]$. In general, a full description of $\sqrt{I}$ can be difficult to obtain. Thankfully, the following result let's us extend the notion of the ideal membership algorithm to check whether a given element is in $\sqrt{I}$, without actually constructing $\sqrt{I}$ itself.

Theorem 3.14. *Let* $I = \langle f_1, \ldots, f_s \rangle$ *be an ideal of* $k[x_1, \ldots, x_n]$ *and let* $f \in k[x_1, \ldots, x_n]$. *Then* $f \in \sqrt{I}$ *if and only if* $1 \in \langle f_1, \ldots, f_s, 1 - yf \rangle$, *an ideal of*

$k[x_1, \ldots, x_n, y]$. *That is, whether* $f \in \sqrt{I}$ *or* $f \notin \sqrt{I}$ *is decidable.*

Why is this useful for us? Later, in Chapter 5, we will want to determine if one algebraic variety is contained in another. Here, an algebraic variety is the set $V(I) = \{(y_1, \ldots, y_n) \in \overline{k}^n \mid f(y_1, \ldots, y_n) = 0, \forall f \in I\}$ for an ideal I of $k[x_1, \ldots, x_n]$. To do so, we need to rely on a cornerstone of algebraic geometry: Hilbert's Nullstellensatz.

When k is algebraically closed, the Nullstellensatz gives us an inclusion-reversing bijective correspondence between algebraic varieties in k^n and radical ideals of $k[x_1, \ldots, x_n]$. In particular, it states that $I(V(J)) = \sqrt{J}$ where J is an ideal of $k[x_1, \ldots, x_n]$, and where $I(U)$ is the ideal of all polynomials that vanish on the set U. We can thus rephrase a question of whether one variety V is contained in another U, as whether the radical ideal corresponding to U is contained in the radical ideal corresponding to V. If we then have defining equations for these radical ideals, then we can use the tools established in this chapter to compute our answer.

CHAPTER 4

MATRIX ALGEBRAS AND LENGTH

When working in the matrix algebra $M_n(k)$, problems are often easier when stated in the language of generating sets. Doing so allows one to work with finite sets. For this reason, with the starting point of a generating set, we are able to computationally decide properties about the algebra $M_n(k)$. Recalling that our representations map into such matrix algebras, this will be immediately be useful. We discuss these concepts for matrix algebras below.

4.1 Generating Sets and Paz's Conjecture

As before, k is a field, and $M_n(k)$ is the algebra of $n \times n$ matrices over k.

Definition 4.1. Let S be a finite subset of $M_n(k)$, containing the identity of $M_n(k)$. We refer to a product of elements of S as a **word** in S. Let $a = a_1 \cdots a_t$ be a word in S; we say t is the **length of** a. By convention, the identity element of $M_n(k)$ has length zero.

Note that a given word a in S may not have a well defined length, in the situation where a can be written as two distinct products of elements. That is, if $a = a_1 \cdots a_t = b_1 \cdots b_s$ where $a_i, b_i \in S$ with $s \neq t$, we cannot canonically assign a length to a. To circumvent this, we may define the notion of a minimal length.

Definition 4.2. Let S be a finite subset of $M_n(k)$, containing the identity of $M_n(k)$. For a word a in S, we define the **minimal length of** a as

$$\min\{t \mid a = a_1 \cdots a_t, \text{ for } a_1, \ldots, a_t \in S\}.$$

Taking S as above, for $i \geq 0$, let S^i denote the set of all words of length less than or equal to i. Additionally, let $\mathrm{Span}(S^i)$ denote the linear span of S^i in $M_n(k)$. Let $\mathrm{Alg}(S)$ be the k-subalgebra generated by S.

Definition 4.3. Let S be a finite subset of $M_n(k)$, containing the identity. The **length of** S is defined as

$$l(S) = \min\{i \in \mathbb{Z}_+ \mid \dim_k(\mathrm{Span}(S^i)) = \dim_k(\mathrm{Alg}(S))\}.$$

The **length of** $M_n(k)$ is then defined as

$$l(M_n(k)) = \max\{l(S) \mid S \text{ is a generating set of } M_n(k)\}.$$

An upper bound on $l(M_n(k))$ depending only on n exists; for example, $l(M_n(k)) \leq n^2$. To showcase this, let S be a generating set of $M_n(k)$. By our definition, $\mathrm{Span}(S^0) = k$. Note that each $\mathrm{Span}(S^i)$ is a k-vector space of $M_n(k)$. This gives an ascending chain of vector spaces

$$k = \mathrm{Span}(S^0) \subseteq \mathrm{Span}(S^1) \subseteq \mathrm{Span}(S^2) \subseteq \cdots .$$

Since $\dim_k(M_n(k)) = n^2$, we know that $\mathrm{Span}(S^{n^2}) = \mathrm{Span}(S^{n^2+1}) = \cdots$. Therefore $l(S) \leq n^2$. Thus we have an upper bound $l(M_n(k)) \leq n^2$.

In [15], Paz conjectured a linear upper bound for $l(M_n(k))$.

Conjecture 4.4. *[15] (Paz's Conjecture) For k a field, $l(M_n(k)) \leq 2n - 2$.*

This conjecture is still open. In the same work by Paz, he proved a non-linear upper bound of $(n^2+2)/3$ for $l(M_n(k))$. In [14] Pappacena was also able to prove a non-linear upper bound, which is currently the best known upper bound, in the general case.

Theorem 4.5. *[14] For k a field, $l(M_n(k)) < n\sqrt{\frac{2n^2}{n-1} + \frac{1}{4}} + \frac{n}{2} - 2$.*

Paz also provided an open lemma which would resolve Paz's conjecture, if proved.

Conjecture 4.6. *[15] Let S be a generating set for $M_n(k)$. If there exists $j \in \mathbb{Z}$ with $j \leq 2n - 2$ such that one of the following holds:*

1. *if $1 \leq j \leq n - 1$ then $\dim_k(\mathrm{Span}(S^j)) - \dim_k(\mathrm{Span}(S^{j-1})) \leq j$,*

2. if $n \leq j \leq 2n-2$ *then* $\dim_k(\mathrm{Span}(S^j)) - \dim_k(\mathrm{Span}(S^{j-1})) \leq 2n-j-2$,

then $l(S) \leq 2n-2$.

In particular, if this holds for every generating set of $M_n(k)$, then of course $M_n(k) \leq 2n-2$. This provides a nice tool for resolving Paz's conjecture for certain classes of generating sets of $M_n(k)$.

Much work has been made to show Paz's conjecture holds in certain cases, which we discuss below.

4.2 Instances of Linear Bounds

Arguably the most substantial result made to date is that Paz's conjecture holds for small n, provided the field k is contained in $\mathbb{C}$.

Theorem 4.7. *[7, 11] For* k *a subfield of* $\mathbb{C}$, *and* $n \leq 6$, $l(M_n(k)) \leq 2n-2$.

For $n \leq 4$ this follows from the fact that $l(M_n(k)) \leq (n^2+2)/3$ as originally shown by Paz. The case of $n = 5$ was shown by Longstaff, Niemeyer, and Panaia in 2006 [11]. The case of $n = 6$ was then shown by Lambrou and Longstaff in 2009 [7]. In particular both papers show the result for pairs of complex matrices. This is especially useful in our case, since we often are computing low-dimensional representations over $\mathbb{Q} \subset \mathbb{C}$.

We now turn our attention to specific generating sets of $M_n(k)$. In general, S will denote a generating set for $M_n(k)$.

Definition 4.8. For k a field and $A \in M_n(k)$, if $\dim_k(\langle 1, A, \ldots, A^{n-1}\rangle) = n$ then A is referred to as **nonderogatory**.

In 2017, Guterman, Laffey, Markova, and Smigoc proved the following results pertaining to generating sets containing nonderogatory matrices:

Theorem 4.9. *[6] For k a field and S a generating set of $M_n(k)$, if some matrix of S is nonderogatory, then $l(S) \leq 2n - 2$.*

Theorem 4.10. *[6] Let be k a field, and S be a generating set of $M_n(k)$.*

1. *If there exists $A \in S$ such that $\deg(A) = n - 1$, then $l(S) \leq 2n - 2$.*

2. *Let $n = 2k$ for $m \in \mathbb{Z}_{\geq 1}$. If $J_m \oplus J_m \in S$, then $l(S) \leq 5n/2 - 2$. Here, J_m is the Jordan matrix.*

In addition to these theorems, Guterman et al. show examples of generating sets S containing a nonderogatory matrix, where the upper bound $l(S) = 2n - 2$ is achieved [6]. This tells us that the bound in Theorem 4.9 is sharp.

The following result by Constantine and Darnall in 2004 shows that Paz's conjecture is satisfied in the presence of a specific property for a generating set [2].

Definition 4.11. [2] Let $S = \{A_1, \ldots A_m\}$ for matrices $A_i \in M_n(k)$, k a field. We say that S has the **modified Poincaré-Birkhoff-Witt (PBW)**

property if every product of elements of S, $u = A_{i_1} \cdots A_{i_l}$ can be written as

$$\sum_{j_1+\cdots+j_t=l} c_{(j_1,\ldots,j_t)} A_t^{j_t} \cdots A_1^{j_1},$$

modulo $\mathrm{Span}(S^{l-1})$, with $c_{(j_1,\ldots,j_t)} = 0$ if $X_t^{j_t} \cdot X_1^{j_1} < u$ in the lexicographical ordering of S.

Theorem 4.12. *[2] For k a field and S a generating set of $M_n(k)$, if S has the modified PBW property, then $l(S) \leq 2n - 2$.*

Constantine and Darnall show the above theorem by showing that for S, as in the theorem, Conjecture 4.6 is true. In addition, they show that if S is not a generating set of $M_n(k)$, and A is the largest subalgebra of $M_n(k)$ k-linearly spanned by all products in S, then products of length at most $2n - 3$ are necessary to k-linearly span A [2].

CHAPTER 5

DECIDABILITY FOR REPRESENTATIONS

With the tools of computational commutative algebra and linear algebra now available, we may examine what properties concerning representations of finitely presented algebras are decidable, at least in low dimensions. The results here are distilled from work by Letzter, in [8, 9, 10].

5.1 General Representations

We begin with a definition, which will be used in following sections as well.

Definition 5.1. Let $n, s \in \mathbb{Z}_{\geq 1}$ and K be a commutative ring. Set

$$B = K[x_{ij}(l) \mid 1 \leq i, j \leq n, 1 \leq l \leq s],$$

$$\boldsymbol{x}_l = \begin{bmatrix} x_{1,1}(l) & \cdots & x_{n,1}(l) \\ \vdots & \ddots & \vdots \\ x_{1,n}(l) & \cdots & x_{n,n}(l) \end{bmatrix} \in M_n(B),$$

for $1 \leq l \leq s$. We call $\boldsymbol{x}_l$ the **l-th generic matrix**. Note that i, j, and l are all indices.

To start, if we have $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t \rangle$ and $A_1, \ldots, A_s \in M_n(\overline{k})$, we can define the map $\rho : R \to M_n(\overline{k})$, with $X_i \mapsto A_i$, for $1 \leq i \leq s$. Then $f_1(A_1, \ldots, A_s) = \cdots = f_t(A_1, \ldots, A_s) = 0$ if and only if ρ is a representation.

Lemma 5.2. *Let $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t \rangle$ be a finitely presented algebra over a computable field k and $n \in \mathbb{Z}_{\geq 1}$. It is decidable whether there exists an n-dimensional representation of R.*

Proof. A representation $\rho : R \to M_n(\overline{k})$ can be determined by the image of the generators, $\big(\rho(X_1), \ldots, \rho(X_s)\big) \in (M_n(\overline{k}))^s$. Therefore there exists a n-dimensional representation of R if and only if the set

$$\{(\Gamma_1, \ldots, \Gamma_s) \in (M_n(\overline{k}))^s \mid f_1(\Gamma_1, \ldots, \Gamma_s) = \cdots = f_t(\Gamma_1, \ldots, \Gamma_s) = 0\}$$

is nonempty. This set is nonempty if and only if the entries of the matrices $f_1(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s), \ldots, f_t(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s)$ have a common zero. This can be determined by a dimension check of the ideal generated by the entries of $f_1(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s), \ldots, f_t(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s)$, which can be computed using Groebner bases. □

5.2 Irreducibility

For $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t\rangle$ and $A_1, \ldots, A_s \in M_n(\overline{k})$, consider the map $\rho : R \to M_n(\overline{k})$, with $X_i \mapsto A_i$, for $1 \leq i \leq s$. As noted, we can determine if ρ is a representation. We can additionally decide if ρ is irreducible. Here, ρ is an irreducible representation if and only if $\mathrm{Alg}(A_1, \ldots, A_s) = M_n(\overline{k})$, if and only if $\dim_k(\mathrm{Alg}(A_1, \ldots, A_s)) = n^2$. The dimension of $\mathrm{Alg}(A_1, \ldots, A_s)$ can be computed using Gaussian elimination.

For the next result about irreducibility, we will need a definition and a theorem from polynomial identity algebras. Below, if $f \in \mathbb{Z}\{Y_1, \ldots, Y_m\}$ and A is a k-algebra, then $f(A)$ will denote the set $\{f(a_1, \ldots, a_m) \mid a_1, \ldots, a_m \in A\}$. A complete treatment of this material can be found in [13].

Definition 5.3. Let $m \in \mathbb{Z}_{\geq 1}$. In $\mathbb{Z}\{Y_1, \ldots, Y_m\}$, let

$$s_m(Y_1, \ldots, Y_m) = \sum_{\sigma \in S_m} \mathrm{sgn}(\sigma) \cdot Y_{\sigma(1)} \cdots Y_{\sigma(m)}.$$

We call s_m the m**-th standard identity**.

Theorem 5.4. *[13] (Amitsur-Levitzky) Let $n \in \mathbb{Z}_{\geq 1}$ and K be a commutative ring. Then $s_m(M_n(K)) = 0$ for all $m \geq 2n$, and $s_m(M_n(K)) \neq 0$ for all $m < 2n$.*

We now present the main result. The algorithm detailed below is implemented into our software package for `SAGE`, and examples of usage are detailed in Chapter 6.

Theorem 5.5. *[8] Let $R = k\langle X_1, \dots, X_s \mid f_1, \dots, f_t\rangle$ be a finitely presented algebra over a computable field k and $n \in \mathbb{Z}_{\geq 1}$. It is a decidable property of R whether there exists an n-dimensional irreducible representation, via the following algorithm:*

```
input: n, require n > 1
input: p, require p > l(M_n(\bar{k}))
begin algorithm
    if not has_representation(n) do
        return False
    end
    B := k[x_ij(l) | 1 ≤ i, j ≤ n, 1 ≤ l ≤ s]
    for 1 ≤ l ≤ s do
        x_l := (x_ij(l)) ∈ M_n(B)
    Rel(B) := the ideal of B generated by the entries of
                  f_i(x_1, ..., x_s), for 1 ≤ i ≤ t
    S := {x_i1 ··· x_im | m < p} = {M_1, ..., M_N}
    U := {M_m0 · s_2n-2(M_m1, ..., M_m2(n-1)) |
               1 ≤ m0 ≤ N, 1 ≤ m1 < ··· < m2n-2 ≤ N}
    for u in U do
        if trace(u) ∉ √Rel(B) do
            return True
```

return False

end

Proof. The set of representations $R \to M_n(\overline{k})$ can be identified with

$$\{(\Gamma_1, \ldots, \Gamma_s) \in (M_n(\overline{k}))^s \mid f_1(\Gamma_1, \ldots, \Gamma_s) = \cdots = f_t(\Gamma_1, \ldots, \Gamma_s) = 0\},$$

which is the algebraic subvariety $V(\mathrm{Rel}(B))$ of $(M_n(\overline{k}))^s$, where $\mathrm{Rel}(B)$ is the ideal generated by the entries of $f_1(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s), \ldots, f_t(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s)$, as described in the algorithm above.

Let P be the set of $(\Gamma_1, \ldots, \Gamma_s) \in (M_n(\overline{k}))^s$ such that the $\overline{k}$-subalgebra generated by $\Gamma_1, \ldots, \Gamma_s$ is not equal to $M_n(\overline{k})$. It can be shown that P and $V(\mathrm{Rel}(B))$ are both algebraic subvarieties of $(M_n(\overline{k}))^s$. Then we have that an irreducible representation of R exists if and only if $V(\mathrm{Rel}(B)) \not\subseteq P$.

Suppose P is defined by equations $g_1 = \cdots = g_q = 0$ in B. Since $\overline{k}$ is algebraically closed, the ideal-variety correspondence given by the Nullstellensatz tells us that $V(\mathrm{Rel}(B)) \not\subseteq P$ if and only if $g_i \notin \sqrt{\mathrm{Rel}(B)}$ for some $1 \leq i \leq q$.

Now let A be a subalgebra of $M_n(\overline{k})$, with a finite generating set of G. Take p to be an upper bound of $l(M_n(\overline{k}))$. Then A is $\overline{k}$-linearly spanned by the set

$$T = \{a_1 \cdots a_i \mid a_1, \ldots, a_i \in G,\ 0 \leq i < p\},$$

where the product corresponding to $i = 0$ is the identity of $M_n(\overline{k})$. Give T an ordering, say $T = \{b_1, \ldots, b_N\}$. Additionally let $L = A \cdot s_{2n-2}(A)$. Then L is

$\overline{k}$-linearly generated, as a left ideal of $M_n(\overline{k})$, by the set

$$V = \{b_{m_0} \cdot s_{2n-2}(b_{m_1}, \dots, b_{m_{2n-2}}) \mid 1 \le m_0 \le N,\, 1 \le m_1 < \cdots < m_{2n-2} \le N\}.$$

There are two cases to consider. For the first case, suppose that A is properly contained in $M_n(\overline{k})$, and let J be the Jacobson radical of A. The algebra A/J will embed into $\bigoplus_1^q M_m(\overline{k})$, for some $m < n$ and some $q \in \mathbb{Z}$. The Amitsur-Levitzky theorem then tells us that $s_{2n-2}(M_m(\overline{k})) = 0$, and further $s_{2n-2}\left(\bigoplus_1^q M_m(\overline{k})\right) = 0$. Thus, $s_{2n-2}(A/J) = 0$. This implies that $s_{2n-2}(A) \subseteq J$. Then L is contained in J, which implies that L is a nilpotent left ideal of A. This further implies that every matrix in L has trace zero.

For the second case, suppose that $A = M_n(\overline{k})$. Then L is a left ideal of $M_n(\overline{k})$, and using a similar argument as above, at least one matrix in L has nonzero trace.

We have shown that A is a proper subalgebra of $M_n(\overline{k})$ if and only if every matrix in L has trace zero. In particular, A is a proper subalgebra of $M_n(\overline{k})$ if and only if $\{\text{trace}(v) \mid v \in V\} = \{0\}$.

Recalling that p is an upper bound for $l(M_n(\overline{k}))$, let

$$S = \{\boldsymbol{x}_{i_1} \cdots \boldsymbol{x}_{i_m} \mid m < p\},$$

and give S an ordering, say $S = \{M_1, \dots, M_N\}$. We then define the set

$$U = \{M_{m_0} \cdot s_{2n-2}(M_{m_1}, \dots, M_{m_{2(n-1)}}) \\ \mid 1 \le m_0 \le N, 1 \le m_1 < \cdots < m_{2n-2} \le N\}.$$

By our results, it follows that $\{\text{trace}(u) = 0 \mid u \in U\}$ is a set of defining equations for P. Therefore an irreducible representation of R exists if and only if $\text{trace}(U) \not\subseteq \sqrt{\text{Rel}(B)}$. From this, the algorithm follows. □

A few remarks on the algorithm described:

1. As noted in Chapter 2, Buchberger's algorithm has double exponential complexity. In the algorithm above, we utilize Buchberger's algorithm for each element of U. Therefore our algorithm has at least double exponential complexity.

2. The size of U depends on the size of S. In particular, if $|S| = q$, then we have $|\,\text{trace}(U)| = |U| = \binom{q}{2n-2}$ elements to apply the radical membership algorithm to. Thus it is wise to attempt to reduce the size of S before applying the algorithm. Items 3 and 4 address this.

3. By the Cayley-Hamilton Theorem, we may express $\boldsymbol{x}_l^n$ as a linear combination of lower powers of $\boldsymbol{x}_l$, for any $1 \leq l \leq s$. Therefore we can exclude terms of S containing $\boldsymbol{x}_l^n$, or any higher power of $\boldsymbol{x}_l$.

4. If the defining relations $f_1 = 0, \ldots, f_t = 0$ of R allow us to rewrite or reduce a monomial $X_{i_1} \cdots X_{i_k}$, we may extend this to the monomial $\boldsymbol{x}_{i_1} \cdots \boldsymbol{x}_{i_k}$. Thus in the presence of satisfactory defining relations, we can drastically reduce the size of S, for large n.

5. One can also look for irreducible representations with restrictions placed on the images of X_l. This can be done by instead assuming that a subset of the generic matrices $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s$ are diagonal, upper triangular, or lower triangular, to reduce computational requirements. Of course, a returned value of `False` in this case does not prevent the existence of an irreducible representation in the more general setting.

Theorem 5.6. *[8] Let $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t\rangle$ be a finitely presented algebra over a computable field k and $n \in \mathbb{Z}_{\geq 1}$. Suppose $k[x]$ is equipped with a factoring algorithm. We can algorithmically produce an n-dimensional representation of R, provided we know the existence of an n-dimensional irreducible representation.*

Proof. Suppose that there exists an n-dimensional irreducible representation of R. Then we can find $y \in \{\text{trace}(u) \mid u \in U\}$ such that $y \notin \sqrt{\text{Rel}(B)}$, using the algorithm described in Theorem 5.5.

Consider the equations given by setting $yz - 1$ and the entries of the matrices $f_1(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s), \ldots, f_t(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s)$ equal to zero, living in the polynomial ring $B[z]$. We can use the factoring algorithm of $k[x]$ to solve these $sn^2 + 1$ equations, giving us a solution $x_{ij}(l) = \lambda_{ij}(l) \in \overline{k}$. Then the map $X_l \mapsto (\lambda_{ij}(l))$ for $1 \leq l \leq s$ gives an irreducible representation of R. □

5.3 Semisimplicity

Letzter also provided algorithms for decisions regarding semisimple representations in [9, 10], similar to those that exist for irreducible representations as described above. We will state the results of this section without proof, and do not provide the algorithms themselves.

Theorem 5.7. *[9] Let R be a finitely presented k-algebra, and $n \in \mathbb{Z}_{\geq 1}$. Whether every m-dimensional representation is semisimple for $m \leq n$ is a decidable property of R.*

Theorem 5.8. *[10] Let k be a computable field of characteristic zero, and R be a finitely presented k-algebra.*

1. *It is decidable whether R has at most finitely many equivalence classes of semisimple n-dimensional representations.*
2. *Suppose $k[x]$ is equipped with a factoring algorithm, and that R has has finitely many equivalence classes of semisimple n-dimensional representations. Then the number of equivalence classes of semisimple n-dimensional representations is computable.*

CHAPTER 6

THE SOFTWARE

`SAGE` is a free, open-source computer algebra system built in `python` and `cython`. For reference on how to install and operate `SAGE`, see [16]. We have developed a package `finitely_presented_algebra` that implements the algorithms for representations as described in Chapter 5. Additionally, the software handles basic functionality and arithmetic for finitely presented algebras. The package is available here:

`https://github.com/rhoadskj/finitely-presented-algebra`

This package is currently not available in the standard distribution of `SAGE`, and must be downloaded and imported separately. The package is written with the intent of following `SAGE` guidelines and traditions as closely as possible, so that users familiar with `SAGE` can use the package without much effort. Any feedback or bug reports may be directed at the author.

What follows in this chapter is an introduction in how to operate the package, as well as examples that make use of the representation-theoretic functions.

6.1 The Basics

Once the `finitely_presented_algebra.py` package has been downloaded, open `SAGE` in the same directory as the file, in order to load the package.

```
sage: from finitely_presented_algebra import *
```

Once loaded, a finitely presented algebra can be formed with the following:

```
sage: R.<x,y> = FinitelyPresentedAlgebra(QQ, 'y*x - 2*x*y'); R
Finitely presented algebra over Rational Field with presentation
<x, y | -2*x*y + y*x>

sage: A = FinitelyPresentedAlgebra(QQ, 'b*a - a*b - 1', 'a, b'); A
Finitely presented algebra over Rational Field with presentation
<a, b | -1 - a*b + b*a>
sage: A.inject_variables()
Defining a, b
```

Note that in the second case, the names for generators must come after the relations, and must be injected to be recognized. Relations and names can be given as a tuple, list, or comma delimited string. Addition and multiplication work as expected.

```
sage: f = R(x + y); g = R(x - y)
sage: f*g
x^2 + x*y - y^2
```

Upon creating an instance of finitely presented algebra, a list of rewrite rules is created, derived from the relations of the algebra and a graded lexicographic ordering. Then when elements are constructed, these rewrite rules are applied when possible.

However, this is currently only implemented for sufficiently 'simple' relations. By this, we mean relations that consist of two or less terms. Thus a monomial is swapped for another monomial, possibly empty, at the cost of multiplication by a coefficient. For example, in `R` and `W` as above:

```
sage: R(y*x + x*y)
3*x*y
sage: A(b*a - a*b)
-a*b + b*a

sage: R(-2*x*y + y*x)
0
sage: A(b*a - a*b - 1)
-1 - a*b + b*a
```

Standard functions for accessing data about the object are included. The functions `ngens()`, `nrels()`, `gen(i)`, `gens()`, `rel()`, and `rels()` can be used to access information about the generators and relations. Additionally, the functions `base_ring()`, `free_algebra()`, `one()`, `zero()`, and `monoid()` can be used to learn underlying properties of the algebra itself. For more detail on use, one can either reference the `README.md` file available on GitHub.

6.2 Algorithm for Representations

In 6.2.1, we detail the basic tests for general representations. Our core work are tests in 6.2.2 and 6.2.3, which perform tests for irreducible representation. In the following sections, all tests were performed on a personal computer, with a 1.70 GHz processor and 8GB of RAM.

6.2.1 Tests for Representation

We saw in Chapter 5 that for a given finitely presented algebra R and $n \in \mathbb{Z}_{\geq 1}$, we can determine if there exists an n-dimensional representation of R, or check if a given map is a representation. We have implemented these tests as `has_rep()`, and `is_rep()`, respectively.

Given a finitely presented algebra $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t\rangle$ and matrices $A_1, \ldots, A_s \in M_n(\overline{k})$, we can check if the map $X_i \mapsto A_i$ for $1 \leq i \leq s$ is an n-dimensional representation using `is_rep()`, giving the list of matrices in an array and the dimension as input. The function then returns `True` if the given map is a representation and `False` otherwise. Note that by assumption the base field of R must be a computable field.

As an example, let $R = \mathbb{Q}_{-1}[X,Y] = \mathbb{Q}\langle X,Y \mid XY = -XY\rangle$, and let $\rho : R \to M_2(\overline{k})$ be the map defined by

$$X \mapsto \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \ Y \mapsto \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

Is ρ a 2-dimensional representation of R? We can check with the following:

```
sage: R.<x,y> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: A1 = matrix([[0, 1], [1, 0]])
sage: A2 = matrix([[0, -1], [1, 0]])
sage: R.is_rep([A1, A2], 2)
True
```

Thus ρ as described above is a representation of R.

As mentioned in Lemma 5.2, we can determine the existence of an n-dimensional representation of R; we do so using `has_rep()`. This function takes input of an integer n, and returns `True` if there exists an n-dimensional representation of R, and `False` otherwise.

In the example of $R = \mathbb{Q}_{-1}[X,Y]$ above, we verified that at least one 2-dimensional representation exists. This matches the result of `has_rep(2)`, as below.

```
sage: R.<x,y,z> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: R.has_rep(2)
True
```

The function `has_rep()` has one optional argument: `restrict`. This argument can be set to a tuple or list of either strings or `None` types. Each string must match `'diagonal'`, `'upper'`, or `'lower'`. The length of `restrict` must match the number of generators of R. This will restrict the i-th generic matrix constructed to being either standard, diagonal, upper triangular, or lower triangular, in accordance with the i-th entry of `restrict`.

As an example, we can ask if $R = \mathbb{Q}_{-1}[X, Y]$ has a representation $\rho : R \to M_n(\overline{k})$ where $\rho(X)$ is upper triangular, and $\rho(Y)$ is lower triangular.

```
sage: R.<x,y,z> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: R.has_rep(2, restrict=['upper', 'lower'])
True
```

6.2.2 Testing if a Representation is Irreducible

Again let $R = k\langle X_1, \ldots, X_s \mid f_1, \ldots, f_t\rangle$ and $A_1, \ldots, A_s \in M_n(\overline{k})$. As noted in Chapter 5, we can check if the map $X_i \mapsto A_i$ for $1 \leq i \leq s$ is an n-dimensional irreducible representation using `is_irred_rep()`, with the same arguments as `is_rep()`, returning `True` if the map is an n-dimensional representation and `False` otherwise. Of course, `is_irred_rep()` calls `is_rep()` to verify that it is working with a representation in the first place, and returns `False` if it is not.

We will use the same example as in 6.2.1, where $R = \mathbb{Q}_{-1}[X, Y]$, and $\rho : R \to M_2(\overline{k})$ is the map

$$X \mapsto \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \; Y \mapsto \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

Knowing that ρ is a representation already, we can check if ρ is irreducible.

```
sage: R.<x,y> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: A1 = matrix([[0, 1], [1, 0]])
sage: A2 = matrix([[0, -1], [1, 0]])
sage: R.is_irred_rep([A1, A2], 2)
True
```

Thus $\rho : R \to M_2(\overline{k})$ as defined above is a 2-dimensional irreducible representation.

6.2.3 Testing Existence of Irreducible Representations

Let R be a finitely presented k-algebra. We stated in Theorem 5.5 that we can decide the existence of an n-dimensional irreducible representation of R. This is implemented in the function `has_irred_rep()`, which, given an input of an integer n, of course returns `True` if there exists an n-dimensional irreducible representation of R, and `False` otherwise.

Again, let $R = \mathbb{Q}_{-1}[X, Y]$. We already have determined the existence of a 2-dimensional irreducible representation of R, which we can also verify using `has_irred_rep()`.

```
sage: R.<x,y,z> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: R.has_irred_rep(2)
True
```

To compare, we can also ask whether there exists a 3-dimensional irreducible representation of R.

```
sage: R.<x,y,z> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: R.has_irred_rep(3)
False
```

The software will attempt to reduce the size of the set S detailed in Theorem 5.5 as much as possible. It does so by removing terms with n-th powers and rewriting terms based on the relations present, similar to how it reduces elements of R. If no 'simple' rewrite rules are present, the generating set can be exceptionally large, preventing the software from practically completing.

The function `has_irred_rep()` has two optional arguments: `restrict`, and `gen_set`. In the right conditions, these arguments can be used to reduce computation time.

The argument `restrict` works equivalently to the similarly named argument of `has_rep()`. Extending the question posed above, does $R = \mathbb{Q}_{-1}[X, Y]$ possess a representation $\rho : R \to M_n(\overline{k})$ where $\rho(X)$ is upper triangular and $\rho(Y)$ is lower triangular?

```
sage: R.<x,y,z> = FinitelyPresentedAlgebra(QQ, 'x*y + y*x')
sage: R.has_irred_rep(2, restrict=['upper', 'lower'])
True
```

The argument `gen_set` can be set to either a set, list, or tuple of elements of `R`, which will be used to create the generating set S after passing the generators to the generic matrices, in lieu of `has_rep()` creating its own generating set. Alternatively, `gen_set` can be set the string `'pbw'` or `'PBW'`, which will create a generating set similar to a Poincaré-Birkhoff-Witt basis. That is, the generating set will be of the form

$$\{\boldsymbol{x}_1^{t_1} \cdots \boldsymbol{x}_s^{t_s} \mid t_i \geq 0, \text{ and } t_1 + \cdots + t_s < p\},$$

with appropiate terms removed by the Cayley-Hamilton theorem. Using `gen_set` is wise in situations where the relations of R do not generate rewrite rules via the software, but a 'nice' basis of the algebra exists.

As an example, the first Weyl algebra $A_1(\mathbb{Q}) = \mathbb{Q}\langle X, Y \mid YX - XY = 1\rangle$ does not produce a 'simple' rewrite rule, by the logic of the software. However, the set $\{X^i Y^j \mid i, j \geq 0\}$ forms a basis of $A_1(\mathbb{Q})$, meaning the option `gen_set='pbw'` can be used freely in order to reduce computational time.

```
sage: A.<x,y> = FinitelyPresentedAlgebra(QQ, 'y*x - x*y - 1')
sage: A.has_rep(3, gen_set='pbw')
False
```

Now that we understand how to use the software and its functions, we will examine the performance times of the algorithms, and then work through specific examples.

6.3 Performance

All tests were performed using the `python` function `timeit()`. All tests were performed on the algebra $A_1(\mathbb{Q}) = \mathbb{Q}\langle X, Y \mid YX - XY = 1\rangle$, unless otherwise stated, to guarantee `False` returns for each function. Additionally, $A_1(\mathbb{Q})$ does not possess a rewrite tool, by the software's understanding, which gives us the largest generating set S possible. The function `is_irred_rep()` was modified to not perform `is_rep()` first to avoid redundancy, and likewise for `has_irred_rep()` and `has_rep()`. All tests were performed with $n = 2$, that is, for 2-dimensional representations.

The function `is_rep(..., 2)` was tested on the map defined by

$$X \mapsto \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}, \qquad Y \mapsto \begin{bmatrix} 3 & 3 \\ 3 & 3 \end{bmatrix}.$$

These matrices were chosen to have nonzero, nonidentity entries. With this, `is_rep(..., 2)` averaged a time of 769 microseconds. Of course, this is our fastest, and simplest, function.

Our function `has_rep(2)` performed with an average time of 1.97 milliseconds. With the additional restriction of `restrict=['upper', 'lower']` as detailed in 6.2.1, the average became 1.89 milliseconds, a 4% increase.

Testing `is_irred_rep(2)` with the same map defined for `is_rep(2)` above, our average time was 13.7 milliseconds. This time can be improved by mapping to matrices with zero or identity entries. As an example, with the map

$$X \mapsto \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad Y \mapsto \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix},$$

our average time was 6.98 milliseconds, which is a 49% increase.

Unsurprisingly, `has_irred_rep()` is our most costly function. In the 2-dimensional case, `has_irred_rep(2)` performed with an average time of 169 milliseconds. When tested on a finitely presented $\mathbb{Q}$-algebra with three generators and no 2-dimensional representations, our function `has_irred_rep(2)` has an average time of 5.93 seconds, which constitutes a 3408% increase. With the restrictions placed of `restrict=['upper', 'lower']`, our average becomes 167 milliseconds. With the option of `gen_set='pbw'`, our average becomes 23 milliseconds, a 86% increase. Of course, this assumption is not applicable to every algebra.

For general finitely presented algebras, testing the existence of irreducible n-representations for $n = 2, 3$ are as much as can expected on a small computer. However, when suitable additional reductions can be applied, somewhat higher values of n can be considered; for instance, see the example regarding $\mathrm{PSL}_2(\mathbb{Z})$ described in 6.4.2.

6.4 Examples

6.4.1 First Weyl Algebra

As discussed in Example 2.7, the first Weyl Algebra over the rationals $A_1(\mathbb{Q}) = \mathbb{Q}\langle X, Y \mid YX - XY = 1\rangle$ has no finite dimensional representations. Thus, our function `has_rep(n)` will return `False` for any value of n, when tested on $A_1(\mathbb{Q})$. We verify this to be the case for $2 \leq n \leq 30$.

```
sage: A.<x,y> = FinitelyPresentedAlgebra(QQ, 'y*x - x*y - 1')
sage: results = {A.has_rep(i) for i in range(2, 31)}
```

```
sage: results
{False}
```

As a additional check, we generate random matrices $M_1, M_2 \in M_2(\overline{\mathbb{Q}})$ and verify that $X \mapsto M_1, Y \mapsto M_2$ does not give a 2-dimensional representation, and repeat this process 100,000 times.

```
sage: A.<x,y> = FinitelyPresentedAlgebra(QQ, 'y*x - x*y - 1')
sage: M = MatrixSpace(QQ.algebraic_closure(), 2)
sage: M = MatrixSpace(QQ.algebraic_closure(), 2)
sage: results = {A.is_rep([M.random_element(), M.random_element()],
....: 2) for i in range(10000)}
sage: results
{False}
```

Thus none of our 100,000 randomly generated maps gave us a representation of $A_1(\mathbb{Q})$, as desired.

6.4.2 Projective Special Linear Group

Our discussion up to now has dealt with representations of associative algebras, but the representation theory of groups is also of great interest.

Definition 6.1. Let G be a group, and k be a field. A **representation of** G **over** k is a group homomorphism $\rho : G \to \mathrm{Aut}_k(V)$, where V is a k-vector space and $\mathrm{Aut}_k(V)$ is the group of k-linear homomorphisms of V. Here, the **dimension of** ρ is the dimension of V as a k-vector space.

There is a natural relation between representations of groups and representations of associative algebras. This comes from the fact that there is a bijective correspondence between representations of the group algebra $k[G]$ and representations of G over k. For a in depth discussion of the representation theory of group algebras, we recommend the reader Chapter 3 of [12].

In our case, if G is a finitely presented, and k is a computable field, then we can use the tests discussed above to gain information about the group representations of G, over k.

To showcase this, consider the projective special linear group $\mathrm{PSL}_2(\mathbb{Z})$. Choosing $\mathbb{Q}$ as our field and letting $n \in \mathbb{Z}_{\geq 1}$, we can ask if there exists an n-dimensional irreducible representation of $\mathrm{PSL}_2(\mathbb{Z})$ over $\mathbb{Q}$. This group is isomorphic to $C_2 * C_3$, where C_m denotes the m-th cyclic group.

Due to this, we can equivalently ask if there exists an n-dimensional irreducible representations of the finitely presented algebra $R = \mathbb{Q}[\mathrm{PSL}_2(\mathbb{Z})] = \mathbb{Q}\langle X, Y \mid X^2 - 1, Y^3 - 1\rangle$, which we can answer computationally. In particular, we want to know the state of existence for irreducible representations of dimension n for $2 \leq n \leq 6$.

```
sage: R.<x,y> = FinitelyPresentedAlgebra(QQ, 'x^2 - 1, y^3 - 1')
sage: results = [R.has_irred_rep(i) for i in range(2, 7)]
sage: results
[True, True, False, False, True]
```

This tells us that for $2 \leq n \leq 6$, there exists an n-dimensional irreducible representation of R if and only if n divides 6. With the bijective correspondence of group representations and group-algebra representations, we learn the same result for n-dimensional irreducible representations of $\mathrm{PSL}_2(\mathbb{Z})$ over $\mathbb{Q}$.

Let us also assume that $\rho(X)$ is diagonal. Doing so, we retain the existence of irreducible representations of dimensions 2, 3, and 6. We check this with the following:

```
sage: R.<x,y> = FinitelyPresentedAlgebra(QQ, 'x^2 - 1, y^3 - 1')
sage: R.has_irred_rep(2, restrict=['diagonal', None])
True
sage: R.has_irred_rep(3, restrict=['diagonal', None])
True
sage: R.has_irred_rep(6, restrict=['diagonal', None])
True
```

6.4.3 Off-diagonal Generating Set

For $n \in \mathbb{Z}_{>1}$ and k a field, consider the following matrices in $M_n(k)$:

$$U_n = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix}, \qquad L_n = \begin{bmatrix} 0 & & & \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix}.$$

If we let p be an upper bound for $l(M_n(k))$, then products of U_n and L_n of length $\leq p$ will always k-linearly span $M_n(k)$. Consequently, if U_n and L_n are contained in a set of matrices S, then S will necessarily k-linearly span all of $M_n(k)$. Using this, we know that if we have a finitely presented k-algebra R, a representation $\rho : R \to M_n(\overline{k})$ is irreducible provided that $U_n, L_n \in \rho(R)$.

Note that $U_n \cdot L_n \cdot U_n = U_n$ for any $n > 1$. Therefore if we define an algebra $R = \mathbb{Q}\langle X, Y \mid XYX - X\rangle$, we know that ρ will be a n-dimensional representation of R. Thus, it will also be an irreducible representation. We verify this for $2 \leq n \leq 6$ using `is_irred_rep()`:

```
sage: A.<a, b> = FinitelyPresentedAlgebra(QQ, 'a*b*a - a')
sage: results = {A.is_irred_rep(off_diag(i), i) for i in range(2, 7)}
sage: results
{True}
```

Thus the result holds for $2 \leq n \leq 6$. The map ρ should be irreducible for all values of n as well, however computation time limits us from checking for large values of n.

REFERENCES

[1] M. Artin. On Azumaya algebras and finite dimensional representations of rings, *J. Algebra*, **11** (1969), 532-563.

[2] D. Constantine, M. Darnall. Lengths of finite dimensional representations of PBW algebras, *Linear Algebra and its Applications*, **395**, (2005), 175-181.

[3] D. Cox, J. Little, D. O'Shea. *Ideals, Varieties, and Algorithms*, 2nd edition, New York, Springer-Verlag, 1997.

[4] T. W. Dubé. The stucture of polynomial ideals and Gröbner bases, *SIAM J. Comput.*, **19(4)**, (1990), 750-773.

[5] K. R. Goodearl, R. B. Warfield, Jr. *An Introduction to Noncommutative Noetherian Rings*, 2nd edition, London, Cambridge, 2004.

[6] A. Guterman, T. Laffey, O. Markova, H. Smigoc. A resolution of

Paz's conjecture in the presence of a nonderogatory matrix. *Linear Algebra and its Applications*, **543** (2018), 234-250.

[7] M. S. Lambrou, W. E. Longstaff. On the lengths of pairs of complex matrices of size six. *Bull. Aust. Math. Soc.*, **80** (2009), 177-201.

[8] E. S. Letzter. Constructing irreducible representations of finitely presented algebras. *J. Symbolic Computation*, **32** (2001), 255-262.

[9] E. S. Letzter, Effective detection of nonsplit module extensions, *J. Pure Appl. Algebra*, **198** (2005), 255-236.

[10] E. S. Letzter, Detecting infinitely many semisimple representations in a fixed finite dimension, *J. Algebra*, **320** (2008), 3926-3934.

[11] W. E. Longstaff, A. C. Niemeyer, Oreste Panaina. On the lengths of pairs of complex matrices of size at most five, *Bull. Aust. Math. Soc.*, **73** (2006), 461-472.

[12] M. Lorenz. *A Tour of Representation Theory*, 1st edition, American Math. Soc. (2018).

[13] J. C. McConnell, J. C. Robson. *Noncommutative Noetherian Rings*, Chichester, John Wiley and Sons, 1987.

[14] C. J. Pappacena. An upper bound for the length of a finite-dimensional algebra. *J. Algebra*, **197** (1997), 535-545.

[15] A. Paz. An application of the Cayley-Hamilton theorem to matrix polynomials in several variables. *Linear Multilinear Algebra*, **15** (1984), 161-170.

[16] W. A. Stein, et al. *Sage Mathematics Software (Version 8.9)*. The Sage Development Team. `http://www.sagemath.org`, 2019.