

Gröbner Bases and Invariant Theory

Michael Hancock

Summer, 2005

Contents

1	Gröbner bases	1
1.1	Preliminaries	1
1.2	Hilbert's Basis Theorem	5
1.3	Monomial Ordering and the Division Algorithm	7
1.4	Basic Gröbner Basis Results	11
1.5	Gröbner Bases and Algebraic Geometry	18
2	Invariant Theory	23
2.1	Preliminaries	23
2.2	The Fundamental Invariant Algorithm	28
2.3	Algorithm for Determining Prime Ideals	34
2.4	Torus Invariants and Integer Programming	41
3	Algebraic Geometry	46
3.1	Noetherian Rings	46
3.2	Algebraic Geometry: Definitions and Basic Results	47
3.3	Computations in Affine Algebraic Sets and k -Algebras	55
3.4	Gröbner Bases and their Varieties	58
3.5	Algebraic Computation and Invariance	62
4	The Road Ahead	70
4.1	Acknowledgements	71
A	Appendix A	72
A.1	Invariant Algorithm	72
A.2	Krull Dimension	84

Abstract

Title: Invariant Semi-Algebraic Geometry

Semi-algebraic geometry studies subsets S of Euclidean space \mathbb{R}^n which are defined by polynomial (inequalities) in $\mathbb{R}[x_1, \dots, x_n]$ (the polynomial ring in n -variables).

For this project, we focus on semi-algebraic sets that display nice symmetries. More precisely, we consider sets that are invariant under the action of some subgroup G of $GL_n(\mathbb{R})$ (the multiplicative group of non-singular $n \times n$ matrices with real coefficients). This is tightly connected to *Invariant Theory*. Via Hilbert's Basis Theorem, we know that the associated ring $\mathbb{R}[x_1, \dots, x_n]^G$ of G -invariant polynomials is finitely generated, say by k generators. This naturally defines the **orbit map** π from \mathbb{R}^n to \mathbb{R}^k . We are interested in analyzing the image $\pi(S)$ of an invariant semi-algebraic set S , under this map π . In particular, we would like to know an answer to the following **two questions**:

- (i) Can $\pi(S)$ be compact, if S is noncompact?
- (ii) Does $\pi(S)$ contain an unbounded cone, if S does?

These questions arose naturally in the context of one of the supervisor's research projects. The resulting work is divided into three broad parts: first, we will go over the theories and techniques necessary for the calculation of Gröbner bases. In the second section the Gröbner bases will be utilised as a means of finding invariant polynomial generators, accompanied with the some background invariant theory. Finally, there will be a brief introduction into algebraic geometry and algebraic sets, culminating in a method of analysing invariant varieties that uses both of the previous sections. Ideally, these results can at some later date be extended to semi-algebraic sets and the questions above.

Chapter 1

Gröbner bases

1.1 Preliminaries

The Gröbner basis is an excellent example of both the power and elegance inherent in abstract algebra. As a tool used for the analysis of polynomial rings, its applications include algorithms for determining whether an ideal is prime, determining whether a given polynomial is contained inside the radical of a polynomial ideal, and, most importantly, determining the generators of an invariant polynomial ring. But before we can really delve into the specifics of the Gröbner basis, we need a brief overview of the underlying concepts behind it.

Definition 1.1.1. A **group** is an ordered pair $(G, *)$ where G is a set and $*$ is a binary operation ($*$ is a binary operation if for each $a, b \in G$, $a * b \in G$) on G satisfying the following axioms:

1. $(a * b) * c = a * (b * c)$ for all $a, b, c \in G$ ($*$ is associative),
2. There exists an element $e \in G$ called the **identity** of G , such that for all $a \in G$ we have $a * e = e * a = a$, and
3. For each $a \in G$ there is an element $a^{-1} \in G$, called the **inverse** of a , such that $a * a^{-1} = a^{-1} * a = e$.

Furthermore, the group $(G, *)$ is called **abelian** (or **commutative**) if $a * b = b * a$ for all $a, b \in G$.

Example 1.1.2. The set of integers, denoted $\mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}$, forms a group under the binary operation of addition, "+". In this case, the identity element is 0 since $x + 0 = 0 + x = x$ for any $x \in \mathbb{Z}$ and the inverse of any element x is $-x$ since $x + (-x) = 0$. The set is associative and closed under addition also hold, so $(\mathbb{Z}, +)$ is a group. The group is also abelian since $a + b = b + a$ for all $a, b \in G$.

Note that the set \mathbb{Z} is not a group under the binary operation of multiplication, \times . It contains an identity element, 1, but no elements except for 1 and -1 have inverses contained within the set. For example, there is no $y \in \mathbb{Z}$ such that $y \times 2 = 1$.

Other common groups include the set of rational numbers, \mathbb{Q} , the set of real numbers, \mathbb{R} , and the set of complex numbers, \mathbb{C} . All three of these sets are groups, under both the operations $+$ and \times .

Example 1.1.3. Another group, one that often appears in invariant theory, is the **symmetry group**, S_n . Consider a set $X = \{x_1, \dots, x_n\}$. A **permutation** of X is a rearrangement of the terms that make up X . The set of all such permutations, or, equivalently, the set of all bijections from X into X , is a group S_X under the operation of function composition, denoted \circ , where first we apply the function on the right side of the \circ , then apply the function on the left. The operation \circ can be shown to be associative and is also a binary operation: $\sigma : X \rightarrow X$ and $\tau : X \rightarrow X$ are both bijections, so $\sigma \circ \tau$ is also a bijection from X to X . The identity of S_X is the permutation 1 that leaves all elements of X exactly as they were originally. For every permutation σ there is an inverse function $\sigma^{-1} : X \rightarrow X$ satisfying $\sigma \circ \sigma^{-1} = 1$. Thus, all the group axioms hold for (S_X, \circ) .

As an example, let (123) represent the element in S_n (which is the symmetry group corresponding to a set X with n elements) that moves x_1 to x_2 , x_2 to x_3 , and x_3 to x_1 ($\{x_1, x_2, x_3\}$ becomes $\{x_2, x_3, x_1\}$). Let (23) represent the permutation that switches x_2 and x_3 . Then the composition (23) \circ (123) is performed by applying the (123) permutation to X to get X' , and then applying the (23) permutation to X' to get a final set, X'' . We have $X = \{x_1, x_2, x_3, x_4, \dots, x_n\}$, $X' = \{x_3, x_1, x_2, x_4, \dots, x_n\}$, and $X'' = \{x_3, x_2, x_1, x_4, \dots, x_n\}$. (In reality, we haven't really changed X at all; $X = X' = X''$. We just use the different notation to demonstrate more clearly how X is affected by the permutations.)

Note that the first permutation moves x_2 to where x_3 was and the second permutation moves x_2 back to its original position, and that under both

permutations, all other elements in the set are unchanged. This means that the net effect of this composition is to switch the first and third elements, and so we have $(23) \circ (123) = (13)$. $(13) \circ (13) = 1$, so (13) is its own inverse. Finally, we know this group is not abelian, since $(123) \circ (23) = (12)$, not (13) .

Definition 1.1.4. A **ring** R is a set together with the two binary operations $+$ and \times , often called addition and multiplication, that satisfy the following axioms:

1. $(R, +)$ forms an abelian group,
2. \times is associative: $(a \times b) \times c = a \times (b \times c)$ for all $a, b, c \in R$, and
3. the distributive laws hold in R ; for all $a, b, c \in R$

$$(a + b) \times c = (a \times c) + (b \times c)$$

and

$$a \times (b + c) = (a \times b) + (a \times c)$$

- i.) The ring R is said to be **commutative** if multiplication is commutative.
- ii.) The ring R is said to have an **identity** if a multiplicative identity exists.
- iii.) A subset $S \subseteq R$ is said to be a **subring** of R if S by itself satisfies the axioms for being a ring.

Note that while (\mathbb{Z}, \times) in Example 1.1.2 was not a group, $(\mathbb{Z}, +, \times)$ is a commutative ring with identity. Often, if it is clear from the context, the \times operation is omitted; instead of $a \times b$, we just use ab .

Definition 1.1.5. A **field** is a commutative ring with identity $1 \neq 0$ such that every nonzero element $a \in R$ has a multiplicative inverse.

Definition 1.1.6. A subset $I \subseteq R$ is said to be an **ideal** of R if I satisfies the following conditions:

1. I is a subring of R .
2. I is closed under multiplication by all elements of R ; for all $r \in R$ and $i \in I$, $ir \in I$ and $ri \in I$.

When actually testing whether a set I is an ideal of a ring R , all we have to test is whether I is non-empty, and if $r(x - y) \in I$. The other conditions are inherited from the ring R .

Definition 1.1.7. Let $A = \{a_1, \dots, a_n\} \subset R$, where A is finite. Then the **ideal finitely generated by A** , denoted (a_1, \dots, a_n) , is the ideal consisting of all elements of the form $\{r_1 a_1 r_1' + \dots + r_n a_n r_n' \mid r_i, r_i' \in R, a_i \in A, n \in \mathbb{Z}^+\}$. Similarly, the **subring finitely generated by A** , denoted (unfortunately) in exactly the same manner is the subring consisting of all elements of the form $\{r_1 a_1 r_1' + \dots + r_n a_n r_n' \mid r_i, r_i', a_i \in A, n, i \in \mathbb{Z}^+\}$. In other words, it's the same as for an ideal, but satisfies the conditions of a subring rather than an ideal. It should generally be clear from the context whether a subring or an ideal is being generated.

Example 1.1.8. Consider the ring \mathbb{Z} . The set $(2) = \{r_1 2 r_1' + \dots + r_n 2 r_n' \mid r_i, r_i', 2 \in \mathbb{Z}\}$ consists of all elements that are multiples of two or sums of multiples of two; in other words, all even numbers. This set is closed under subtraction for (2) and closed under multiplication for all \mathbb{Z} , so it is an ideal generated by 2. In fact, for any $n \in \mathbb{Z}$, (n) forms an ideal in \mathbb{Z} generated by n .

Definition 1.1.9. Fix a commutative ring R with identity. Let x be an indeterminate. The formal sum $a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ with $n \geq 0$ and each $a_i \in R$ is a **polynomial** in x with coefficients $a_i \in R$. If $a_n \neq 0$ then the polynomial is said to be of **degree n** , $a_n x^n$ is called the **leading term**, and a_n is called the **leading coefficient**. The set of all possible polynomials of this form is called the **polynomial ring** in the variable x with coefficients in R , and is denoted $R[x]$.

Note that $R \subset R[x]$, as the set of all constant polynomials. Multiplication and addition on the polynomial ring are defined as the usual multiplication and division of polynomials.

Definition 1.1.10. Any nonzero polynomial in x_1, \dots, x_n indeterminates with coefficients in R is a finite sum of **monomial terms**, which are terms of the form $a x_1^{d_1} x_2^{d_2} \dots x_n^{d_n}$, where $a \in R$ is called the **coefficient** of the term, and the d_i are nonnegative integers. If $a = 1$, the monomial term is said to be **monic**. The sum $d = d_1 + \dots + d_n$ is called the **degree** of the term, and the ordered n -tuple (d_1, \dots, d_n) is the **multidegree** of the term. The **degree** of a nonzero polynomial is the largest degree of any of its monomial

terms. A multivariable polynomial that has the same degree for all terms is said to be **homogeneous**. For example, $x^3 + xyz + y^2z + z^3$ is a homogeneous polynomial of degree 3.

Definition 1.1.11. The **multivariable polynomial ring** can be formed with a ring R and $\{x_1, \dots, x_n\}$ indeterminates. It is defined as the set of all polynomials that are finite sums of elements of the form $ax_1^{d_1}x_2^{d_2}\dots x_n^{d_n}$ where $a \in R$ and $d_i \in \mathbb{Z}^+$, the set of positive integers. $R[x_1, \dots, x_n]$ can be thought of as $R[x_1, \dots, x_{n-1}][x_n]$, the ring of polynomials with x_n as an indeterminate and the coefficients are polynomials in $n - 1$ variables.

This section was a short list of the terms and concepts underlying the basic principles of both Gröbner bases and invariant theory. It is meant to serve as a brief review, and is by no means a comprehensive look at these topics. For a more in-depth approach, consult [6].

1.2 Hilbert's Basis Theorem

Definition 1.2.1. A commutative ring R with 1 is called **Noetherian** if every ideal of R is finitely generated (that is, if every ideal $I \subset R$ can be expressed as $I = (a_1, \dots, a_n)$, $a_1, \dots, a_n \in R$).

Theorem 1.2.2. *Hilbert's Basis Theorem* ([6], p 316) (*1-Variable Case*)

If R is a Noetherian ring then so is the polynomial ring $R[x]$.

Proof. Let I be an ideal in $R[x]$ and let L be the set of all *leading coefficients* (the term of each polynomial sum that contains the highest degree of x in that sum) in I .

Claim 1: L is an ideal of R .

As mentioned in the previous section, we must show L is non-empty, closed under addition for all elements in L , and closed under multiplication for all $R[x]$. First, $0 \in I$, so L is non-empty. Let $ax^d + \dots = f(x)$ and $bx^e + \dots = g(x)$ be polynomials in I of degrees e and d with leading coefficients in R . Then $ax^d, bx^e \in L$. Let $r \in R$. Then either $r - ab = 0$ or it is the leading coefficient of the polynomial $rx^e f - x^d g$. Since this polynomial is in

I , by virtue of I being closed under addition and under all multiplication, $ra - b \in L$, and so L is an ideal of R .

Since R is assumed to be Noetherian, the ideal L must be finitely generated. Arbitrarily, let $L = (a_1, \dots, a_n), a_i \in R$. Let f_i be an element of I whose leading coefficient is a_i . Let e_i denote the degree of f_i , and let N be the maximum of e_1, \dots, e_n . For each degree $d \in 0, 1, \dots, N-1$, let L_d be the set of all leading coefficients of polynomials in I of degree d , along with 0.

Claim 2: L_d is an ideal of R .

$0 \in L_d$ by definition. Let $f(x) = ax^d + \dots$ and $g(x) = bx^d + \dots$ be in I such that $a, b \in L_d$. Then $f(x) - g(x) = (a - b)x^d + \dots$ so $a - b \in L_d$. For any $r \in R$, $rf(x) = rax^d + \dots \in I$, so $ra \in L_d$. Thus, L_d is an ideal of R .

R is finitely generated (since it is an ideal of $R[x]$), which means L_d is also finitely generated. For each nonzero ideal L_d , let $b_{d,1}, \dots, b_{d,n} \in R$ be a set of generators for L_d and let $f_{d,i}$ be a polynomial in I of degree d with leading coefficient $b_{d,i}$.

Claim 3: $I = (\{f_1, \dots, f_n\} \cup \{f_{d,i} \mid 0 \leq d < N, 1 \leq i \leq n_d\})$. Let $I' = (\{f_1, \dots, f_n\} \cup \{f_{d,i} \mid 0 \leq d < N, 1 \leq i \leq n_d\})$. $I' \subseteq I$, since all the generators of I' were chosen to be in I . If $I' \neq I$, then there exists a nonzero polynomial $f \in I$ of minimum degree with $f \notin I'$. Let $d = \deg f$ and let a be the leading coefficient of f .

Suppose $d \geq N$. Since $a \in L$, we may write a as an R -linear combination of the generators of L : $a = r_1a_1 + \dots + r_na_n$. Then $g = r_1x^{d-e_1}f_1 + \dots + r_nx^{d-e_n}f_n$ is an element of I' with the same degree d and leading coefficient a as f . Then $f - g \in I$ is a polynomial in I of smaller degree than f but $f - g$ is not in I' , because then $f = f - g + g$ would be in I' . Because f was the smallest nonzero polynomial in I but not in I' , $f - g = 0$. But then $f = g, f \in I'$, which is a contradiction.

Now suppose $d < N$. Then $a \in L_d$ for some $d < N$, and we may write $a = r_1b_{d,1} + \dots + r_nb_{d,n}$ for some $r_i \in R$. Then $g = r_1f_{d,1} + \dots + r_nf_{d,n}$ is a polynomial in I' with the same degree and leading coefficient a as f , and so $f - g \in I \neq I'$. By minimality, $f - g = 0, f = g$, and we have a contradiction as before. \square

Thus, for any ideal I in $R[x]$, we know that there exists a finite set of generators $\{f_1, \dots, f_n\}$ such that $I = (f_1, \dots, f_n)$. By an inductive argument, the same reasoning extends to multivariable polynomial rings:

(Multi-variable Case)

Every ideal in the polynomial ring $R[x_1, \dots, x_n]$ with coefficients from a Noetherian ring R is finitely generated.

Proof. This follows from previous results, where we stated that $R[x_1, \dots, x_n] = R[x_1, \dots, x_{n-1}][x_n]$. Inductively, it follows that $R[x_1, \dots, x_{n-1}][x_n] = (R[x_1, \dots, x_{n-2}][x_{n-1}])[x_n] = \dots = ((\dots (R[x_1])[x_2]) \dots)[x_n]$, which we know to be finitely generated by Hilbert's Theorem in the 1-Variable Case. \square

This result means that every ideal in a multivariable polynomial ring has a finite set of generators. Given such a generating set, we can then calculate the ideal's Gröbner basis.

1.3 Monomial Ordering and the Division Algorithm

When selecting the leading term of a polynomial in the single variable case, it is fairly easy to tell, for example, that $x^7 > x^4$. But what is the leading term in a multivariable polynomial like $xy + x^2 + y^2$? The answer depends on the monomial ordering. But before we get to monomial ordering, we need a few other ordering definitions.

Definition 1.3.1. A **total ordering** (or a *linear ordering*) on a set S is any relation, typically denoted \leq , where the following holds for all $a, b, c \in S$:

- if $a \leq b$ and $b \leq a$ then $a = b$ (antisymmetry),
- if $a \leq b$ and $b \leq c$ then $a \leq c$ (transitivity), and
- In particular, for all $a, b \in S$, either $a \leq b$ or $b \leq a$. (totalness)

Definition 1.3.2. A **well-ordering** on a set S is a *total ordering* on S with the property that every non-empty subset S has a least element in this ordering.

Definition 1.3.3. A **monomial ordering** is a well ordering " \geq " on the set of monomials that satisfies $mm_1 \geq mm_2$ whenever $m_1 \geq m_2$ for monomials m, m_1, m_2 . Given a monomial ordering and a polynomial $f = m_1 + m_2 + \dots + m_n$, where m_i is a monomial for $1 \leq i \leq n$, the **leading term** of f is the monomial m_j such that $m_j \geq m_i$, for all i , under the monomial ordering. The leading term of f is often denoted $LT(f)$.

Although the conditions on a monomial ordering may seem like a fairly simple at first glance, the requirement that it be a well ordering is actually very restrictive, and its other property restricts the monomial ordering even further.

Definition 1.3.4. Given a multivariable polynomial ring $R[x_1, \dots, x_n]$, declare an ordering of variables, for example $x_1 > x_2 > \dots > x_n$. Then the monomial term $Ax_1^{a_1} \dots x_n^{a_n}$ with the multidegree (a_1, \dots, a_n) has higher order than the monomial term $Cx_1^{c_1} \dots x_n^{c_n}$ with the multidegree (c_1, \dots, c_n) if the first component where the multidegrees differ has $a_i > c_i$. It can be shown that this ordering defines a monomial ordering. We call this ordering the **lexicographic monomial ordering** on the monomials of the ring $R[x_1, \dots, x_n]$.

Proposition 1.3.5. *The lexicographic monomial ordering is a monomial ordering.*

Proof. Let $m_1 = Ax_1^{a_1} \dots x_n^{a_n}$, $m_2 = Bx_1^{b_1} \dots x_n^{b_n}$, $m_3 = Cx_1^{c_1} \dots x_n^{c_n}$ all be monomials in $R[x_1, \dots, x_n]$, with multidegrees (a_1, \dots, a_n) , (b_1, \dots, b_n) , (c_1, \dots, c_n) and degrees a, b, c , respectively. Assume that $m_1 \geq m_2$. That means $a_i > b_i$, where $1 \leq i \leq n$, $b_j = a_j$ for $j < i$. Multiply both m_1 and m_2 by m_3 . Then m_1m_3 has multidegree $(a_1 + c_1, \dots, a_n + c_n)$, and m_2m_3 has multidegree $(b_1 + c_1, \dots, b_n + c_n)$. For $j < i$, it's clearly true that $a_j + c_j = b_j + c_j$. And for i , it is also clear that $a_i + c_i > b_i + c_i$, since we can get the relation $a_i > b_i$ by subtracting c_i from both sides of the inequality. Thus, $m_1m_3 \geq m_2m_3$, fulfilling one of the conditions for a monomial ordering. The second condition follows from the fact that lexicographic monomial ordering hinges on the ordering of the multidegree. Since the elements of the n -tuple multidegree are elements of \mathbb{N} , and $<$ is known to form a well-ordering on the set of \mathbb{N} , the lexicographic monomial ordering is a well-ordering as well. \square

Definition 1.3.6. Given a multivariable polynomial ring, we impose the following ordering: the monomial term $Ax_1^{a_1} \dots x_n^{a_n}$ with multidegree (a_1, \dots, a_n) has higher order than the monomial term $Cx_1^{c_1} \dots x_n^{c_n}$ with multidegree (c_1, \dots, c_n) when the degree $d = a_1 + \dots + a_n$ is greater than the degree $c = c_1 + \dots + c_n$ or, if if the two degrees are equal, when the first term is lexicographically greater than the other, where lexicographic is defined as

above. This ordering can also be shown to be a monomial ordering, commonly called the **total degree ordering**, which assigns priority to the term with the greatest number of exponents, then to the lexicographic order.

Proposition 1.3.7. *The total degree ordering is a monomial ordering.*

Proof. Let m_1, m_2, m_3 be as in 1.3.5. Let $m_1 \geq m_2$. Then $a > b$ for the sum of the multidegrees, or $a = b$ and m_1 is lexicographically greater than m_2 . We have already proven that the lexicographic monomial ordering is a monomial ordering, so assume $a > b$. Then $m_1 m_3$ has multidegree $(a_1 + c_1, \dots, a_n + c_n)$, which means its degree is

$$\sum_{i=1}^n a_i + c_i = \sum_{i=1}^n a_i + \sum_{i=1}^n c_i = a + c.$$

Likewise, the degree of $m_2 m_3$ is $b + c$. In that case, it's clear that $a + c > b + c$, because we can just subtract c from both sides to get the assumed inequality $a > b$, which means that $m_1 m_3 \geq m_2 m_3$, satisfying that condition of a monomial ordering. The other condition, that the ordering be a well-ordering, follows in a manner similar to the proof of Proposition 1.3.5. The total degree ordering is characterized first by the monomials' degree, which is a sum of natural numbers, making it a natural number. Thus, the total degree ordering is a well-ordering because the set of natural numbers \mathbb{N} is a well-ordering under " $>$ ". The total degree ordering's secondary condition, which is applied when the two monomial have identical degree, is the lexicographic monomial ordering, which has already been shown to be a well-ordering. Thus, the total degree ordering is a monomial ordering. \square

Example 1.3.8. *Monomial Orderings*

1. Consider the polynomial given earlier, $xy + x^2 + y^2$. If we define a lexicographic monomial ordering $x > y$, then the leading term of the polynomial is x^2 . If we define an lexicographic monomial ordering by $y > x$, then the leading term of the polynomial is y^2 .
2. Consider the monomial set $\{x^2, xy^2, z^7, xyz, x^3y, xz^3, y^2\}$. If we define a lexicographic monomial ordering $x > y > z$, then the same set, ordered from highest to lowest, is $\{x^3y, x^2, xy^2, xyz, xz^3, y^2, z^7\}$. If instead we use $x > z > y$, then the ordered set from highest to lowest is $\{x^3y, x^2, xz^3, xyz, xy^2, z^7, y^2\}$. Finally, if we use a total degree ordering $x >$

$y > z$, then the ordered set from highest to lowest is $\{z^7, x^3y, xz^3, xy^2, xyz, x^2, y^2\}$.

Armed with our knowledge of monomial ordering and leading terms, we can now perform general polynomial division with the following algorithm.

Algorithm 1.3.9. General Polynomial Division

(Adapted from [6], p 320)(One should probably take note that the b_i 's in the below algorithm represent polynomials, whereas the b_i 's in the above propositions represented the multidegree components of monomials, and take care not to confuse the two.)

1. Input: monomial ordering on $F[x_1, \dots, x_n]$, a set of nonzero polynomials $b_1, \dots, b_m, f \in F[x_1, \dots, x_n]$, where b_1, \dots, b_m are a set of divisors to be applied to f .
2. Set q_1, \dots, q_m, r to initially be 0 and let $t := 1$.
3. If $LT(b_t)$ divides $LT(f)$, then go to Step 4. Otherwise, let $t := t + 1$. If $t > m$, then go to Step 5. Repeat while $t \leq m$.
4. (a) Find the quotient a_t such that $LT(f) = a_t LT(b_t)$.
 (b) set $q_t := q_t + a_t$
 (c) set $f := f - a_t b_t$.
 If $f = 0$, END. Otherwise, set $t := 1$ and go to Step 3.
5. (a) Set $r := r + LT(f)$
 (b) Set $f := f - LT(f)$.
 If $f = 0$, END. Otherwise, set $t := 1$ and go to Step 2.
6. Output: set of polynomial quotients q_1, \dots, q_m and remainder r such that

$$f = q_1 b_1 + \dots + q_m b_m + r \tag{1.3.1}$$

This algorithm expresses f in two components, one contained in the ideal $I = (b_1, \dots, b_m)$ and the other a remainder r , where $r \notin I$ (Unless $r = 0$). Note that r is not unique and depends on the order of the divisors b_1, \dots, b_m .

Example 1.3.10. 1. Fix the lexicographic ordering $y > x$ on $F[x, y]$. Let $f(x, y) = x^3y^3 + 3x^2y^4$, $b(x, y) = xy^4$. Problem: express $f(x, y)$ in terms of b . $LT(f) = 3x^2y^4$, $LT(b) = xy^4$, so $LT(b) \mid LT(f)$; (In this context, " \mid " is shorthand for "divides".) $f = 3x(b)$, $q = 3x$. Set $f = x^3y^3$, and then $LT(f)$ is not divisible by $LT(b)$, so we set $r = x^3y^3$. Subtracting this from f gives 0, so we finish with $f = 3x(xy^4) + x^3y^3$. Note that the lexicographic ordering $x > y$ gives the same result.

2. Under the same ordering, let $f = y^2 - y - x^2 - x$, $b_1 = xy + 1$, $b_2 = y + x$, and again try to express f , but now in terms of b_1 and b_2 . Then $LT(b_1) = xy$, $LT(b_2) = y$. This means that whenever f is divisible by b_1 , it'll also be divisible by b_2 , which suggests there's going to be multiple decompositions for f . Anyway, for the first round of decompositions, $LT(f) = y^2$, which means $LT(b_2) \mid LT(f)$. $LT(f) = LT(b_2)y$, so we set $q_2 = y$ and $f = y^2 - y - x^2 - x - y^2 - xy = -xy - y - x^2 - x$. $LT(f) = -xy$, $LT(b_1) \mid LT(f)$. Set $q_1 = -1$, $f = -xy - y - x^2 - x + xy + 1 = -y - x^2 - x + 1$. Now $LT(f) = -y$, $LT(b_1) \mid LT(f)$, and so we set $q_2 = y - 1$, $f = -y - x^2 - x + 1 + y + x = -x^2 + 1$. Neither leading term divides $-x^2$ or 1, so we set $r = -x^2 + 1$ and terminate the process. $f = (-1)(xy + 1) + (y - 1)(x + y) - x^2 + 1$.
3. Consider the problem directly above, but after we set $q_2 = y$, $f = -xy - y - x^2 - x$, choose the b_2 factor rather than the b_1 factor as the divisor. Then we set $b_2 = y - x$, $f = -xy - y - x^2 - x + xy + x^2 = -y - x$, which is clearly divisible by b_2 , and we finish with $f = (y - x - 1)(x + y)$. This shows that the remainder is not unique. The next section will go over the conditions necessary for the algorithm to produce a unique remainder for a given set of divisors and a polynomial f .

1.4 Basic Gröbner Basis Results

Definition 1.4.1. Fix a monomial ordering on $R = F[x_1, \dots, x_n]$. If I is an ideal in $F[x_1, \dots, x_n]$, then the **ideal of leading terms**, $LT(I)$, is the ideal generated by the leading terms of all the elements in the ideal: $LT(I) = (LT(f) \mid f \in I)$.

Definition 1.4.2. A **Gröbner basis** for an ideal I in the polynomial ring $F[x_1, \dots, x_n]$ is a finite set of generators b_1, \dots, b_m for I whose leading terms generate the ideal of all leading terms in I :

$$I = (b_1, \dots, b_m) \text{ and } LT(I) = (LT(b_1), \dots, LT(b_m)).$$

The most important property of the Gröbner bases is that if a set $\{b_1, \dots, b_m\}$ is a Gröbner basis, then the remainder r obtained from Algorithm 1.3.9 is unique.

Theorem 1.4.3. (*[6], p 321*)

Fix a monomial ordering on $R = F[x_1, \dots, x_n]$ and suppose $\{b_1, \dots, b_m\}$ is a Gröbner basis for the nonzero ideal I in R . Then

1. *Every polynomial $f \in R$ can be written uniquely in the form*

$$f = f_I + r$$

where $f_I \in I$ and no nonzero monomial term of the 'remainder' r is divisible by any of the leading terms $LT(b_1), \dots, LT(b_m)$.

2. *Both f_I and r can be computed by general polynomial division by b_1, \dots, b_m and are independent of the order in which these polynomials are used in the division.*
3. *The remainder r provides a unique representative for the coset of f in the quotient ring $F[x_1, \dots, x_n]/I$. In particular, $f \in I$ if and only if $r = 0$.*

Proof. 1. Setting $f_I = \sum_{i=1}^m q_i b_i \in I$ in the general polynomial division of f by b_1, \dots, b_m gives the decomposition $f = f_I + r$ for any generators b_1, \dots, b_m . We proceed to prove that this decomposition is unique. Suppose the set $\{b_1, \dots, b_m\}$ is a Gröbner basis, and $f = f_I + r = f_I' + r'$. Then $r - r' = f_I - f_I' \in I$, which means its leading term is in $LT(I) = LT(b_1, \dots, b_m)$, by the definition of the Gröbner basis. Every element in this ideal is a sum of multiples of the monomial terms $LT(b_1), \dots, LT(b_m)$, which means every element is a sum of monomial terms and each term is divisible by some $LT(b_i)$. But r, r' are sums of polynomial terms which are not divisible by any $LT(b_i)$, because any term divisible by a $LT(b_i)$ would have been factored out of them by the Division Algorithm. That means $r - r'$ contains no terms divisible by any $LT(b_i)$ either, which is impossible unless $r - r' = 0$, and then $r = r', f_I = f_I'$, which shows the uniqueness of the two polynomials, proving Statement 1.

2. From the Division Algorithm, we know that f and r can be computed algorithmically. The uniqueness of the Division Algorithm implies that r is independent of the order of b_1, \dots, b_m , and further implies that $f_I = \sum_{i=1}^m q_i b_i$ is unique, although the individual quotients q_i may not be. This gives Statement 2.
3. The uniqueness of r follows immediately from the proof of Statement 1. For the second part of the statement, if $r = 0$, $f = f_I \in I$. If $f \in I$, then $f = f_I + 0$ together with the uniqueness of r implies that $r = 0$, completing the proof. □

Recall 3) of Example 1.3.10. Now, let $I = (xy + 1, y + x)$. We know from our calculations that $f = y^2 - y - x^2 - x = (-1)b_1 + (y - 1)b_2 - x^2 + 1 = (y - x - 1)b_2$. The last decomposition shows clearly that $f \in I$, but the second shows that there exists decompositions of f where $r \neq 0$. This proves that if $\{b_1, \dots, b_m\}$ is a generating set for the ideal I but not a Gröbner basis, then the final statement of 3) in Theorem 1.4.3 is true only in the \leftarrow direction; to complete the other direction, it is necessary that the set under consideration be a Gröbner basis.

Theorem 1.4.4. *Fix a monomial ordering on $F[x_1, \dots, x_n]$ and let I be a nonzero ideal in $F[x_1, \dots, x_n]$. Then:*

1. *If g_1, \dots, g_m are any elements of I such that $LT(I) = (LT(b_1), \dots, LT(b_m))$, then $\{b_1, \dots, b_m\}$ is a Gröbner basis for I .*
2. *The ideal I has a Gröbner basis.*

Proof. 1. We need to show that the set $\{b_1, \dots, b_m\}$ satisfies the properties of being a Gröbner basis. We assume it satisfies the property that $LT(I) = (LT(b_1), \dots, LT(b_m))$, so all that remains to be shown is that the set generates the ideal I . Suppose the contrary, that there exists an $f \in I$ such that f is not generated by the set $\{b_1, \dots, b_m\}$. In that case, by the Division Algorithm, f can be expressed as $f = \sum_{i=1}^m q_i b_i + r$, where no nonzero term in the remainder r is divisible by any $LT(b_i)$. Since $f \in I$, $r \in I$, and so $LT(r) \in LT(I)$. Then $LT(r)$ is divisible by some $LT(b_i)$, which is a contradiction unless $r = 0$. Hence, $f = \sum_{i=1}^m q_i b_i$, and $\{b_1, \dots, b_m\}$ generate I , making the set a Gröbner basis and proving Statement 1.

2. To prove the second statement, we need to show that there always is a set of polynomials in I that satisfies the conditions of being a Gröbner basis. The ideal $LT(I)$ is a monomial ideal, generated by the leading terms of all polynomials in I . By Theorem 1.2.2, a finite number of such leading terms are sufficient to generate $LT(I)$, say $LT(I) = (LT(h_1), \dots, LT(h_k))$ for some $h_1, \dots, h_k \in I$. By Statement 1, this is sufficient to show that the set $\{h_1, \dots, h_k\}$ is a Gröbner basis for I . □

This proves the existence of a Gröbner basis. In order to prove uniqueness, we must tighten our restrictions.

Definition 1.4.5. Fix a monomial ordering on $R = F[x_1, \dots, x_n]$. A Gröbner basis $\{b_1, \dots, b_m\}$ for the nonzero ideal I in R is called a **reduced Gröbner basis** if

- a.) each b_i has monic leading term ($LT(b_i)$ has 1 for a leading coefficient for all $i = 1, \dots, m$), and
- b.) no term in b_j is divisible by $LT(b_i)$ for $j \neq i$.

Theorem 1.4.6. (*[6], p 326*) Fix a monomial ordering on $R = F[x_1, \dots, x_n]$. Then there is a unique (up to a given monomial ordering) reduced Gröbner basis for every nonzero ideal I in R .

Proof. First, consider the claim that two reduced bases have identical leading terms and the same number of elements. If this statement was not true, the set of leading terms would generate different $LT(I)$ ideals, which is impossible if both are supposed to generate I . Once we accept this, the proof becomes a typical uniqueness proof. Let $G = \{g_1, \dots, g_m\}$ and $H = \{h_1, \dots, h_m\}$ be two reduced Gröbner bases of the nonzero ideal I . Then after a possible rearrangement, $LT(g_i) = LT(h_i) = k_i$ for $i = 1, \dots, m$. For any fixed i , consider $f_i = g_i - h_i$. If f_i is nonzero, then since $f_i \in I$, its leading term must be divisible by some k_j . By definition, $k_j, j \neq i$, does not divide any terms in either g_i or h_i , and hence does not divide $LT(f_i)$. But k_i also does not divide $LT(f_i)$ since all terms in f_i have a strictly smaller multidegree. thus, $f_i = 0$ and $g_i = h_i$, so $G = H$. This proves the uniqueness of the reduced Gröbner basis. □

The uniqueness of the Gröbner basis, coupled with the property that its leading terms generates the ideal of leading terms in I allows it to be used in a wide variety of applications. The applications usually depend on being able to explicitly determine the Gröbner basis, which in turn requires Buchberger's Algorithm.

Notation 1.4.7. Let f_1, f_2 be two polynomials in $F[x_1, \dots, x_n]$ and let M be the monic least common multiple of $LT(f_1)$ and $LT(f_2)$. Then we define $S(f_1, f_2)$ as follows:

$$S(f_1, f_2) = \frac{M}{LT(f_1)}f_1 - \frac{M}{LT(f_2)}f_2 \quad (1.4.1)$$

First, a preliminary lemma necessary for the proof of the following theorem.

Lemma 1.4.8. *Suppose $f_1, \dots, f_m \in F[x_1, \dots, x_n]$ are polynomials with the same multidegree α and that the linear combination $h = a_1f_1 + \dots + a_mf_m$ with constants $a_i \in F$ has strictly smaller degree. Then*

$$h = \sum_{i=2}^m c_i S(f_{i-1}, f_i)$$

for some constants $c_i \in F$.

Proof. This proof is basically a matter of expanding the sum given above. Write $f_i = c_i f_i'$, where $c_i \in F$ and f_i' is a monic polynomial of multidegree α . Then $h = \sum a_i c_i f_i' = a_1 c_1 (f_1' - f_2') + (a_1 c_1 + a_2 c_2)(f_2' - f_3') + \dots + (a_1 c_1 + \dots + a_{m-1} c_{m-1})(f_{m-1}' - f_m') + (a_1 c_1 + \dots + a_m c_m) f_m'$. Note that $f_{i-1}' - f_i' = S(f_{i-1}, f_i)$. Then since h and each $f_{i-1}' - f_i'$ has multidegree strictly smaller than α , we have $a_1 c_1 + \dots + a_m c_m = 0$, so the last term on the right hand side is 0, and the lemma follows. Thus, linear combinations of differences account for all cancellation in leading terms of polynomials of the same multidegree. In other words, the difference $S(f_1, f_2)$ is of a multidegree less than the least common multiple of f_1 and f_2 . \square

Theorem 1.4.9. *(Buchberger's Criterion) ([6], p 324) Let $R = F[x_1, \dots, x_n]$ and fix a monomial ordering on R . If $I = (b_1, \dots, b_m)$ is a nonzero ideal in R , then $B = \{b_1, \dots, b_m\}$ is a Gröbner basis for I if and only if $S(b_i, b_j) \in I$, $1 \leq i, j \leq m$, $i \neq j$.*

Proof. First, we assume the set is a Gröbner basis and work to show $S(b_{i-1}, b_i) = 0$. If $\{b_1, \dots, b_m\}$ is a Gröbner basis for I , then $S(b_i, b_j) \in I$, and so $S(b_{i-1}, b_i) = 0$ by Statement 3 of 1.4.3. Next, we assume $S(b_i, b_j) = 0$ for $1 \leq i < j \leq m$. To prove that the set $\{b_1, \dots, b_m\}$ is a Gröbner basis, we need to show that the collection of the set's leading terms generates $LT(I)$, which is the same as showing that $LT(f) \in (LT(b_1), \dots, LT(b_m))$ for an arbitrary $f \in I$. Since $f \in I$, we can write it as $f = \sum_{i=1}^m h_i b_i$ for some polynomials h_1, \dots, h_m . This representation is not unique, so we choose the one for which the largest multidegree of any summand ($\max_{i=1, \dots, m} \delta(h_i b_i)$) is minimal, say α . Then the multidegree of f is no larger than the largest multidegree of all the summands $h_i b_i$, so $\delta(f) \leq \alpha$. We write

$$\begin{aligned} f &= \sum_{i=1}^m h_i b_i = \sum_{\delta(h_i b_i) = \alpha} h_i b_i + \sum_{\delta(h_i b_i) < \alpha} h_i b_i = \\ &\sum_{\delta(h_i b_i) = \alpha} LT(h_i) b_i + \sum_{\delta(h_i b_i) = \alpha} (h_i - LT(h_i)) b_i + \sum_{\delta(h_i b_i) < \alpha} h_i b_i \end{aligned}$$

Suppose that $\delta(f) < \alpha$. Then since the multidegree of the second two sums is also strictly smaller than α it follows that the multidegree of the first sum is strictly smaller than α . If $a_i \in F$ denotes the constant coefficient of the monomial term $LT(h_i)$ then $LT(h_i) = a_i h_i'$ where h_i' is a monomial. By the previous theorem, we can write the first sum as $\sum c_i s(h_{i-1}' b_{i-1}, h_i' b_i)$ with $\delta(h_{i-1}' b_{i-1}) = \delta(h_i' b_i) = \alpha$. Let β_{i-1} be the multidegree of the monic least common multiple of $LT(b_{i-1})$ and $LT(b_i)$. Then $S(h_{i-1}' b_{i-1}, h_i' b_i) = S(b_{i-1}, b_i)$ multiplied by the monomial of multidegree $\alpha - \beta_{i-1, i}$. $S(b_{i-1}, b_i)$ has multidegree less than $\beta_{i-1, i}$ and by assumption, $S(b_{i-1}, b_i) \equiv 0 \pmod{B}$. This means that after division of $S(b_{i-1}, b_i)$ by b_1, \dots, b_m , each $S(b_{i-1}, b_i)$ can be written as a sum $\sum q_j b_j$ with $\delta(q_j b_j) < \beta_{i-1, i}$. It follows that each $S(h_{i-1}' b_{i-1})$ is a sum $\sum q_j' b_j$ with $\delta(q_j' b_j) < \alpha$. But then all the sums on the right hand side of the equation can be written as a sum of terms of the form $p_i b_i$, where p_i satisfies $\delta(p_i b_i) < \alpha$. This contradicts the minimality of α which means $\delta(f) = \alpha$. Then our equation becomes $LT(f) = \sum_{\delta(h_i b_i) = \alpha} LT(h_i) LT(b_i)$ so indeed $LT(f) \in (LT(b_1), \dots, LT(b_m))$, and thus $\{b_1, \dots, b_m\}$ is a Gröbner basis. \square

This proposition, combined with Algorithm 1.3.9 allows us to create an algorithm for determining Gröbner bases.

Algorithm 1.4.10. Buchberger's Algorithm ([6], p 324)

1. Input: $B = \{b_1, \dots, b_m\}$, where $I = (b_1, \dots, b_m)$.
2. Set $i := 1, j := 1$.
3. Calculate $S(b_i, b_j)$, then implement Algorithm 1.3.9, with $f = S(b_i, b_j)$. If $r \neq 0$, go to step 4. Otherwise, go to step 5.
4. Append $b_{m+1} = r$ to B so that $B = \{b_1, \dots, b_m, b_{m+1}\}$. Set $i := 1, j := 1, m := m+1$, and go to step 3.
5. Let $j := j + 1$. If $j \leq m$, then go to step 3. Otherwise, let $j := 1, i := i + 1$. If $i \leq m$ then go to step 3. Otherwise, go to step 6.
6. Set $k := 1$.
7. Let $b_k := \frac{1}{LC(b_k)}(b_k)$, where $LC(b_k)$ denotes the leading coefficient of b_k . Let $k := k + 1$. Repeat while $k \leq m$. If $k > m$, then go to Step 8.
8. Apply Algorithm 1.3.9, with $b_k = f$ and use $B - \{b_k\}$ as the dividing set. After, set $b_k = r$.
9. (a) If $f \neq r$, set $k := 1$ and go to Step 8.
(b) Let $k := k+1$. If $k \leq m$, go to Step 8. If $k > m$, then END.
10. Output: A reduced Gröbner basis, G .

Steps 1 through 5 create a Gröbner basis. Step 7 alters the basis to conform with the first condition of a reduced Gröbner basis, while step 8 alters it to conform with the second condition.

Example 1.4.11. a) Consider the ideal $I = (x^3y - xy^2 + 1, x^2y^2 - y^3 - 1)$ under the lexicographic ordering $x > y$. Test whether $B = \{f_1, f_2\}$ is a Gröbner basis. Recall that

$$S(f_1, f_2) = \frac{M}{LT(f_1)}f_1 - \frac{M}{LT(f_2)}f_2 = S_{1,2} \quad (1.4.2)$$

Making the preliminary calculations, $LT(f_1) = x^3y, LT(f_2) = x^2y^2, M = (x^3y^2)$, so $M/LT(f_1) = y, M/LT(f_2) = x$. $S(f_1, f_2) = y(f_1) - x(f_2) = x^3y^2 - xy^3 + y - x^3y^2 + xy^3 + x = y + x$. $LT(S_{1,2}) = x$, which is not divisible by any of the leading terms, so $S_{1,2} \neq 0 \pmod{(f_1, f_2)}$, and thus the set is not a Gröbner basis.

b) Consider the ideal $I = (x^3y - xy^2 + 1, x^2y^2 - y^3 - 1, x + y)$ under the lexicographic ordering $x > y$. Test whether $B = \{f_1, f_2, f_3\}$ is a Gröbner basis. $S_{1,2}$ is clearly 0, as the one term we needed to reduce it to zero has been added to the set. Now, we have to calculate $S_{1,3}$ and $S_{2,3}$ to check if they are also zero. $LT(f_3) = x$, and $M = x^3y$, so $S_{1,2} = f_1 - x^2yf_3 = x^3y - xy^2 + 1 - x^3y - x^2y^2 = -x^2y^2 - xy^2 + 1$. Now we apply the division algorithm. $LT(S_{1,2})$ is divisible by x , so we set $q_3 = -xy^2$ and let $S_{1,2} = S_{1,2} - (-xy^2)(f_3) = -x^2y^2 - xy^2 + 1 + x^2y^2 + xy^3 = xy^3 - xy^2 + 1$. The new expression is equal to f_1 , so we reduce it by f_1 and achieve the desired result, $S_{1,2} = 0$. Next we check $S_{1,3}$. $M = x^2y^2$, so $S_{1,3} = f_2 - xy^2f_3 = x^2y^2 - y^3 - 1 - x^2y^2 - xy^3 = -xy^3 - y^3 - 1$. $LT(S_{1,3})$ is divisible by x , so we set $q_3 = -y^3$ and $S_{1,3} = S_{1,3} - (-y^3)f_3 = -xy^3 - y^3 - 1 + xy^3 + y^4 = y^4 - y^3 - 1$. This result is not divisible by any leading term in the set, so this set is also not a Gröbner basis. Consider the set that $\{x^3y - xy^2 + 1, x^2y^2 - y^3 - 1, x + y, y^4 - y^3 - 1\}$. This set gives the desired results for $S_{1,2}$, $S_{1,3}$ and $S_{2,3}$, and further calculation will show that it also satisfies all the other differences, so it is a Gröbner basis for I .

Proposition 1.4.12. *Let I and J be two ideals in $F[x_1, \dots, x_n]$. Then $I = J$ if and only if I and J have the same reduced Gröbner basis with respect to any fixed monomial ordering on $F[x_1, \dots, x_n]$.*

The previous proposition follows by the uniqueness of the reduced Gröbner basis. The result is a simple example of the power of the Gröbner bases; often, it's difficult to tell just by looking at their generators if two ideals are identical, but Gröbner bases provide a computational means to do just that.

1.5 Gröbner Bases and Algebraic Geometry

This section provides a brief glimpse into algebraic geometry, albeit only as a possible application of Gröbner bases. Suppose $S = \{f_1, \dots, f_m\}$ is a collection of polynomials in n variables x_1, \dots, x_n , and we are trying to find the values of the form (a_1, \dots, a_n) for the x_1, \dots, x_n such that $f_i(a_1, \dots, a_n) = 0$ for all i , $1 \leq i \leq m$. If (a_1, \dots, a_n) is a solution to this system of polynomials, then it is also a solution of any element f of the ideal $I = (S)$. Equally important, if $S' = \{h_1, \dots, h_s\}$ is another generating set of the ideal I , then

the set of solutions such that $h_1 = \dots = h_s = 0$ is the same as the set of solutions for the original set S . Finding the set of solutions is a very fundamental question in algebraic geometry, and we're going to look at how Gröbner bases can be applied to do just that.

Definition 1.5.1. If I is an ideal in $F[x_1, \dots, x_n]$, then $I_i = I \cap F[x_{i+1}, \dots, x_n]$ is called the *i -th elimination ideal* of I with respect to the ordering $x_1 > \dots, x_n$.

Theorem 1.5.2. *Elimination Theorem* ([6], p 328) Suppose B is a Gröbner basis for the nonzero ideal I in $F[x_1, \dots, x_n]$ with respect to the lexicographic monomial ordering $x_1 > \dots > x_n$. Then $B \cap F[x_{i+1}, \dots, x_n]$ is a Gröbner basis of the i -th elimination ideal, $I_i = I \cap F[x_{i+1}, \dots, x_n]$ of I . In particular, $I \cap F[x_{i+1}, \dots, x_n] = 0$ if and only if $B \cap F[x_{i+1}, \dots, x_n] = \emptyset$.

Proof. Denote $B_i = B \cup F[x_{i+1}, \dots, x_n]$. Then $B_i \subseteq I_i$, since $B \subset I$. To prove it is a Gröbner basis of I_i , it suffices to show that $LT(B_i)$, the set of leading terms of the elements of B_i , generates $LT(I_i)$ as an ideal in $F[x_{i+1}, \dots, x_n]$. Since $B_i \subseteq I_i$, it follows that $LT(B_i) \subseteq LT(I_i)$, where both are ideals in $F[x_1, \dots, x_n]$. To show the reverse containment, we must prove that any arbitrary element $LT(f) \in LT(I_i)$ can be generated by the set $(LT(B_i))$, where $f \in I_i$. Since $f \in I_i$, it follows that $f \in I$, and since B is a Gröbner basis for I , we have $LT(f) = a_1(x_1, \dots, x_n)LT(b_1) + \dots + a_m(x_1, \dots, x_n)LT(b_m)$ for some polynomial $a_i \in F[x_1, \dots, x_n]$ (I realise there's room for some notational ambiguity in using a_i to denote polynomials so soon after using it to denote points in the introduction to this section, but keep in mind a_i is being considered strictly as a polynomial throughout this proof). Writing each a_i as a sum of monomial terms, we reduce $LT(f)$ to a sum of monomial terms of the form $ax_1^{s_1} \dots x_n^{s_n} LT(b_i)$. Since $LT(f)$ involves only the variables x_{i+1}, \dots, x_n , the sum of the terms containing x_1, \dots, x_i must be zero, and so $LT(f)$ is the sum of the monomial terms that only use the variables x_{i+1}, \dots, x_n . $LT(f)$ can be written as a $F[x_{i+1}, \dots, x_n]$ -linear combinations of some monomial terms $LT(b_t)$ where $LT(b_t)$ does not involve the variables x_1, \dots, x_i . But by the choice of ordering, if $LT(b_t)$ doesn't involve x_1, \dots, x_i , neither do the other terms in b_t , so $b_t \in B_i$. Hence $LT(f)$ can be written as $F[x_{i+1}, \dots, x_n]$ -linear combination of elements $LT(B_i)$. Thus, $LT(B_i) = LT(I_i)$, and thus $\cup F[x_{i+1}, \dots, x_n]$ is a Gröbner basis of the i -th elimination ideal, I_i .

To prove the final statement, first assume that $I \cap F[x_{i+1}, \dots, x_n] = 0$. Since $B \in I$, but 0 is not in B , $B \cup F[x_{i+1}, \dots, x_n] = \emptyset$. Conversely, assume

$B \cup F[x_{i+1}, \dots, x_n] = \emptyset$. Then since I is generated by B , the only element that is, and must, be in both I and $F[x_{i+1}, \dots, x_n]$ is 0, completing the proof. \square

Elimination ideals are abstractly interesting, but it may not be immediately apparent how they can be used in finding solution sets. The process works as follows. If the set S defined previously is composed only of linear polynomials, we know we can find the solution set using linear algebra. For nonlinear polynomials, matters are slightly more complicated, and depend on a rather rigid set of conditions. Suppose there is a nonzero polynomial in the ideal I generated by S that involves only one of the variables, say $p(x_n)$. Then the last coordinate a_n is a solution of $p(x_n) = 0$. If there is a polynomial in I involving only x_{n-1} and x_n , say $q(x_{n-1}, x_n)$, then the coordinate a_{n-1} is a solution to $q(x_{n-1}, a_n) = 0$. If we can successively find polynomials in I that eliminate the variables x_1, \dots, x_n , then we will be able to determine all solutions (a_1, \dots, a_n) to our original system of equations explicitly. Finding equations that follow from the system of equations in S , finding elements of the ideal I that do not involve some of the variables, is known as *elimination theory*, and that is where we get the name *elimination ideal*. So, basically, the process of finding a solution set for S boils down to finding the Gröbner basis for the ideal I generated by S , which is guaranteed to have the same solution set, hoping that the basis works out nicely in such a way that its elimination ideals are appropriate for finding the solutions under some ordering, and solving the resulting simpler equations. Some examples follow to illustrate this procedure.

Example 1.5.3. 1. Solve the system of equations $x^2 - yz = 3$, $y^2 - xz = 4$, $z^2 - xy = 5$ over \mathbb{C} . We begin by calculating the Gröbner basis of the ideal $I = (x^2 - yz - 3, y^2 - xz - 4, z^2 - xy - 5)$, under the arbitrarily-chosen lexicographic monomial ordering $x > y > z$. Eventually, we find that $B = \{13x + 11z, 13y - z, -169 + 36z^2\}$, which is ideal for applying our elimination technique. $-169 + 36z^2 = 0 \rightarrow 36z^2 = 169 \rightarrow z^2 = 169/36 \rightarrow z = 13/6, -13/6$. Choosing $z = -13/6$, we get $13y - z = 0 \rightarrow y = z/13 \rightarrow y = (-13/6)/13 = -1/6$ and $13x + 11z = 0 \rightarrow x = -11z/13 \rightarrow x = -11(-13/6)/13 = 11/6$. Choosing $z = 13/6$, we get $y = -1/6$ and $x = -11/6$. Thus, the set of solutions for the system is $(-11/6, 1/6, 13/6)$ and $(11/6, -1/6, -13/6)$.

2. Find a Gröbner basis for the ideal $I = (x^2 + xy + y^2 - 1, x^2 + 4y^2 - 4)$

for the ordering $x > y$ and use it to find the four points of intersection of the ellipse $x^2 + xy + y^2 = 1$ with the ellipse $x^2 = 4y^2 - 4$ in \mathbb{R}^2 . The Gröbner basis is $\{-22y^2 + 9 + 13y^4, -13y^3 + 13y + 3x\}$, which is a bit more computationally unpleasant than our previous question. Applying the quadratic equation yields that $y^2 = 1$ or $9/13$, which means y has the possible values of $1, -1, 3/\sqrt{13}, -3/\sqrt{13}$. Substituting those values into the second equation yields the various values for x , which are $0, 0, 10/\sqrt{13}, -10/\sqrt{13}$, respectively. Thus, the intersection occurs at $(0, 1), (0, -1), (3/\sqrt{13}, 10/\sqrt{13}), (-3/\sqrt{13}, -10/\sqrt{13})$.

3. Use Gröbner bases to find all six solutions to the system of equations $2x^3 + 2x^2y^2 + 3y^3 = 0$ and $3x^5 + 2x^3y^3 + 2y^5 = 0$ over \mathbb{C} . The Gröbner basis of $I = (2x^3 + 2x^2y^2 + 3y^3, 3x^5 + 2x^3y^3 + 2y^5)$ is $\{-4680y^9 + 10872y^8 - 1728y^{10} + 384y^{12} + 1552y^{11} + 6305y^7, 35349800xy^5 + 5069952y^{11} + 22457520y^{10} - 15328944y^9 - 68143848y^8 + 140162180y^7 + 119305575y^6, 318148200x^2y^3 + 56490624y^{11} + 232061040y^{10} - 247553728y^9 - 750245976y^8 + 1576944360y^7 + 1285848975y^6 - 141399200y^5, 2x^3 + 2x^2y^2 + 3y^3\}$. Setting the first polynomial to equal zero and solving for y yields the terms $0, -1/2, -5/2, -97/24, 3/2 + i, 3/2 - i$. The corresponding x values are $0, 1/2, -5/2, -97/36, 1 - 3/2i, 1 + 3/2i$, respectively, and the solution set is $\{(0, 0), (1/2, -1/2), (-5/2, -5/2), (-97/36, -97/24), (1 - 3/2i, 3/2 + i), (1 + 3/2i, 1 - 3/2i)\}$.

We conclude the section with an application of elimination ideals, brought about by considering the generators for combinations of ideals. Let $I = (f_1, \dots, f_s)$ and $J = (h_1, \dots, h_t)$ both be ideals in the polynomial ring $R[x_1, \dots, x_n]$. Then, following straight from the definition of the product and sum of ideals, $I + J = (f_1, \dots, f_s, h_1, \dots, h_t)$ and $IJ = (f_1h_1, f_1h_2, \dots, f_1h_t, f_2h_1, \dots, f_sh_t)$. The generators of the intersection of the two ideals, $I \cap J$, require a bit more work to derive.

Proposition 1.5.4. *If I and J are two ideals in $F[x_1, \dots, x_n]$, then $tI + (1-t)J$ is an ideal in $F[t, x_1, \dots, x_n]$ and $I \cap J = (tI + (1-t)J) \cap F[x_1, \dots, x_n]$. In particular, $I \cap J$ is the first elimination ideal of $tI + (1-t)J$ with respect to the ordering $t > x_1 > \dots > x_n$.*

Proof. Since I and J are ideals in $F[x_1, \dots, x_n]$, they are also ideals in $F[t, x_1, \dots, x_n]$. And since I and J are ideals in $F[t, x_1, \dots, x_n]$, so are

tI and $(1-t)J$, which further implies that the sum $tI + (1-t)J$ is an ideal in $F[t, x_1, \dots, x_n]$, proving the first part of the first statement. If $f \in I \cap J$, then $f = tf + (1-t)f \in tI + (1-t)J \cap F[t, x_1, \dots, x_n]$, proving $I \cap J$ is contained in $tI + (1-t)J \cap F[x_1, \dots, x_n]$. Conversely, suppose $f = tf_1 + (1-t)f_2 \in F[x_1, \dots, x_n]$, where $f_1 \in I$, $f_2 \in J$. then $f = t(f_1 - f_2) + f_1 + f_2 \in F[x_1, \dots, x_n]$, and $t(f_1 - f_2) \in F[x_1, \dots, x_n]$, and so $f_1 - f_2 = 0 \rightarrow f_1 = f_2, f = f_1 = f_2, f \in I \cap J$. The final statement follows from the definition of the elimination ideal. \square

Thus, we have $tI + (1-t)J = (tf_1, \dots, tf_s, (1-t)h_1, \dots, (1-t)h_t)$ if I and J are generated as before. By Theorem 1.5.2, the elements not involving t in the Gröbner basis for the ideal in $F[t, x_1, \dots, x_n]$ computed for the lexicographic monomial ordering $t > x_1 > \dots > x_n$ give a Gröbner basis for the ideal $I \cap J$ in $F[x_1, \dots, x_n]$. A brief example will illustrate this principle.

Example 1.5.5. Use Gröbner bases to compute the intersection of the ideals $I = (x^3y - xy^2 + 1, x^2y^2 - y^3 - 1)$ and $J = (x^2 - y^2, x^3 + y^3)$ in $F[x, y]$. $(tI + (1-t)J) = (t(x^3y - xy^2 + 1), t(x^2y^2 - y^3 - 1), (1-t)(x^2 - y^2), (1-t)(x^3 + y^3))$. Under the lexicographic monomial ordering $t > x > y$, we obtain the Gröbner basis $B = \{xy^2 + y^3, x^2 - y^2, y^4t - ty^3 - t, xt + yt\}$. Then $(G) \cap R[x_1, \dots, x_n] = I \cap J = (xy^2 + y^3, x^2 - y^2)$. To check our results, we let $I = (x^2 - y^2, x^3 + y^3)$, $J = (x^3y - xy^2 + 1, x^2y^2 - y^3 - 1)$ and use the lexicographic monomial ordering $t > y > x$ to compute the Gröbner basis of $(t(x^2 - y^2), t(x^3 + y^3), (1-t)(x^3y - xy^2 + 1), (1-t)(x^2y^2 - y^3 - 1))$. This turns out to be $B = \{xy^2 + y^3, x^2 - y^2, y^4t - ty^3 - t - y^4 + y^3 + 1, xt + yt - x - y\}$, and so $I \cap J = (xy^2 + y^3, x^2 + y^2)$, just as before.

Chapter 2

Invariant Theory

2.1 Preliminaries

Like the Gröbner basis, Invariant Theory has many applications, which include integer programming and projective geometry. Overall, the most common use of invariant theory is simplification; many problems that can't be solved in a general case can be reduced to a simpler invariant case, which can then be used to analyze the general problem. (see [10], [8]) However, for the purpose at hand, rather than focus on the applications of invariant theory, we will focus on invariant theory as an ends rather than a means, finishing with the use of the Gröbner basis to obtain a set of primary invariants.

First, we require some basic invariant theory background and terminology.

Definition 2.1.1. For each $n \in \mathbb{N}$ let $GL_n(F)$ be the set of all $n \times n$ matrices whose entries come from a field F and whose determinant is nonzero:

$$GL_n(F) = \{A \mid A \text{ is an } n \times n \text{ matrix with entries from } F \text{ and } \det(A) \neq 0\}$$

The binary operation associated with $GL_n(F)$ is matrix multiplication. The set is closed under this operation, since if $\det(AB) = \det(A)\det(B)$, where $\det(A) \neq 0$ and $\det(B) \neq 0$, then $\det(AB) \neq 0$, and so $AB \in GL_n(F)$. If $\det(A) \neq 0$, then it is a well-known linear algebra property that A has a multiplicative inverse, so that group condition is also satisfied. The identity is the $n \times n$ identity matrix, where every non-diagonal element is 0 and every diagonal element is 1. Thus, $GL_n(F)$ forms a group under matrix multiplication.

Definition 2.1.2. A **group action** of a group G on a set A is a map from $G \times A$ to A (written as $g \cdot a$ for all $g \in G$ and $a \in A$) satisfying the following properties:

1. $g_1 \cdot (g_2 \cdot a) = (g_1 g_2) \cdot a$, for all $g_1, g_2 \in G$, $a \in A$, and
2. $1 \cdot a = a$ for all $a \in A$, where 1 is the identity of the group G .

Example 2.1.3. Let $G = GL_2(F)$ be a group acting on the ring $F[x, y]$ such that for any $f(x, y) \in F[x, y]$, $A \cdot f(x, y) = f(\theta_1, \theta_2)$, where if $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in G$, then $A \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ax + by \\ cx + dy \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$. This satisfies the conditions for a group action, since $1 \cdot f = f$, and if $B = \begin{bmatrix} r & s \\ t & u \end{bmatrix}$ and A equals as previously, then $AB = \begin{bmatrix} ar + bt & as + bu \\ cr + dt & cs + du \end{bmatrix}$ and $A \cdot (B \cdot f(x, y)) = A \cdot f(rx + sy, tx + uy) = f(arx + asy + btx + buy, crx + csy + dtx + duy) = (AB) \cdot f(x, y)$. For a more concrete example, let $C = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Then C acts on a general polynomial $f(x, y) \in F[x, y]$ by $C \cdot f(x, y) = f(y, x)$. Note that this concept of a group action can be generalized to any $n \in \mathbb{N}$, not just $n = 2$.

Definition 2.1.4. In the most general sense, an **invariant** is a quantity that remains unchanged under certain classes of transformations. For example, the real numbers are invariant under complex conjugation. In terms of group actions, an set A being acted on by a group G is said to be **G-invariant**, or **invariant under the group action** if $g \cdot A = \{g \cdot a_1, \dots, g \cdot a_n\} = \{a_1, \dots, a_n\} = A$ for all $g \in G$; that is, if g applied to every element in the set A ultimately yields the same set A , for every element g in the group G . An individual element $a \in A$ is said to be **invariant** if $g \cdot a = a$ for all $g \in G$.

Now that we have the basic definition of invariancy, we can look at the most familiar example of invariant polynomials, polynomials that are invariant under the group action S_n , as defined in Example 1.1.3.

Definition 2.1.5. A polynomial $f \in \mathbb{C}[x_1, \dots, x_n]$ is said to be **symmetric** if it is invariant under every permutation of the variables x_1, \dots, x_n . For example, $f_1 = x_1 x_2 x_3 + x_2^3 + x_1^3 + x_3^3$ is symmetric, but $f_2 = x_1 x_2 x_3 + x_2^3 + x_1^3$ is not, since $f_2(x_1, x_2, x_3) \neq f_2(x_3, x_2, x_1)$.

Theorem 2.1.7. (*[13], p4*) Let the polynomial $p_k(x_1, \dots, x_n) = x_1^k + x_2^k + \dots + x_n^k$ be known as the ***k-th power sum***. The ring of symmetric polynomials is generated by the first n power sums:

$$\mathbb{C}[x_1, \dots, x_n]^{S_n} = \mathbb{C}[\sigma_1, \dots, \sigma_n] = \mathbb{C}[p_1, \dots, p_n].$$

Proof. In essence, the proof consists of an algorithm that takes a polynomial expressed in terms of the symmetric polynomials and turns it into a polynomial expressed in terms of the n power sums. Then the other necessary traits are inherited from the fact that the symmetric polynomials are a fundamental system of invariants. \square

While the symmetric polynomials are perhaps the best known topic in algebraic invariant theory, we still need a little more background before we can consider other invariant sets.

One of the problems arising in group actions is that it is often difficult to demonstrate explicitly how elements of a group act on elements of a set. This is especially a problem in invariant theory, where we often wish to know which polynomials in a given polynomial ring are invariant. Often, we choose the group G in such a way that it can be represented as a subgroup of the general linear group, and then we can explicitly find the action of G on a polynomial ring by calculating the results of the equivalent matrix group acting on the ring, as in Example 2.1.3.

Definition 2.1.8. Let G be a finite group, let F be a field. A **matrix representation** of G is any homomorphism from G into $GL_n(F)$. (A **homomorphism** is a mapping ρ from a group (G, \diamond) into a group (H, \clubsuit) such that $\rho(x) \clubsuit \rho(y) = \rho(x \diamond y)$ for all $x, y \in G$.)

Not every group has a matrix representation, and some groups are represented by infinite matrix groups. We will focus on groups that are known as **reductive** groups, which have the property that they can be represented by a finite matrix group. In particular, all finite groups are reductive groups.

Proposition 2.1.9. Let G be a reductive group acting on a polynomial ring $\mathbb{C}[x_1, \dots, x_n]$. Let $\mathbb{C}[x_1, \dots, x_n]^G$ denote the set of all polynomials which are invariant under G . Then $\mathbb{C}[x_1, \dots, x_n]^G$ is a subring in $\mathbb{C}[x_1, \dots, x_n]$.

Proof. $\mathbb{C}[x_1, \dots, x_n]^G$ is nonempty, since $0 \in \mathbb{C}[x_1, \dots, x_n]^G$ (In fact, $\mathbb{C} \subset \mathbb{C}[x_1, \dots, x_n]^G$; every constant polynomial is invariant under a matrix group).

If $f, h \in \mathbb{C}[x_1, \dots, x_n]^G$, then $f + h \in \mathbb{C}[x_1, \dots, x_n]^G$, since $g \cdot (f + h) = g \cdot f + g \cdot h = f + h$ for all $g \in G$. If $f \in \mathbb{C}[x_1, \dots, x_n]^G$, $-f \in \mathbb{C}[x_1, \dots, x_n]^G$, since $-f = -1 \times f$, and both f and -1 are invariant under the group action. All of the other necessary properties follow transitively from $\mathbb{C}[x_1, \dots, x_n]$. \square

Note that while $\mathbb{C}[x_1, \dots, x_n]^G$ is a subring, it is usually not an ideal, since it is usually not closed under multiplication for all of $\mathbb{C}[x_1, \dots, x_n]$.

Definition 2.1.10. The **Reynolds operator** * is a mapping defined as follows for a finite matrix group G acting on an polynomial ring $\mathbb{C}[x_1, \dots, x_n]$:

$$\begin{aligned} * : \mathbb{C}[x_1, \dots, x_n] &\rightarrow \mathbb{C}[x_1, \dots, x_n]^G, \\ f &\mapsto f^* := \frac{1}{|G|} \sum_{g \in G} f \circ g. \end{aligned}$$

The Reynolds operator observes the following properties:

1. * is a \mathbb{C} -linear map: $(\lambda f + \nu h)^* = \lambda f^* + \nu h^*$ for all $f, h \in \mathbb{C}[x_1, \dots, x_n]$ and $\lambda, \nu \in \mathbb{C}$.
2. * restricts to the identity map on $\mathbb{C}[x_1, \dots, x_n]^G$: $h = h^*$ for all invariants $h \in \mathbb{C}[x_1, \dots, x_n]^G$.
3. $(fh)^* = f^*h$ for all $f \in \mathbb{C}[x_1, \dots, x_n]^G, h \in \mathbb{C}[x_1, \dots, x_n]$.

Theorem 2.1.11. Hilbert's Finiteness Theorem (*[13] p 26*) *The invariant ring $\mathbb{C}[x_1, \dots, x_n]^G$ of a finite matrix group $G \subset GL(\mathbb{C}^n)$ is finitely generated.*

Proof. Let $I_G := (\mathbb{C}[x_1, \dots, x_n]_+^G)$ be the ideal in $\mathbb{C}[x_1, \dots, x_n]$ generated by all homogeneous invariants of positive degree. By Definition 2.1.10, every invariant h is a \mathbb{C} -linear combination of symmetrized (made to be invariant) monomials $(x_1^{e_1} \dots x_n^{e_n})^*$. These homogeneous invariants are the images of monomials under the Reynolds operator. This implies that the ideal I_G is generated by the polynomials $(x_1^{e_1} \dots x_n^{e_n})^*$, where $e = (e_1, \dots, e_n)$ ranges over all non-zero, nonnegative integer vectors. By Theorem 1.2.2, every ideal in the polynomial ring $\mathbb{C}[x_1, \dots, x_n]$ is finitely generated, which means there exists finitely many homogeneous invariants h_1, \dots, h_m such that $I_G = (h_1, \dots, h_m)$. If we can prove that all the homogeneous invariants $h \in \mathbb{C}[x_1, \dots, x_n]^G$ can be expressed purely in terms of h_1, \dots, h_m , then we have proven $\mathbb{C}[x_1, \dots, x_n]^G$ is finitely generated.

Suppose that there exists some h , where h is a homogeneous element of minimum degree in $\mathbb{C}[x_1, \dots, x_n]^G$, that can't be expressed as a function of $\{h_1, \dots, h_m\}$. Since $h \in I_G$, we have $I = \sum_{j=1}^s f_j h_j$ for some homogeneous polynomials $f_j \in \mathbb{C}[x_1, \dots, x_n]$ of degree less than $\deg(h)$. Applying the Reynolds operator on both sides of this equation yields:

$$h = h^* = \left(\sum_{j=1}^s f_j h_j \right)^* = \sum_{j=1}^s f_j^* h_j \quad (2.1.1)$$

The new coefficients f_j^* are homogeneous invariants whose degree is less than $\deg(I)$. From the minimality assumption on h we get $f_j^* \in \mathbb{C}[h_1, \dots, h_m]$ and therefore $h \in \mathbb{C}[h_1, \dots, h_m]$, which is a contradiction to our assumption. Thus, all homogeneous invariants can be expressed as algebraic combinations of $\{h_1, \dots, h_m\}$. \square

This theorem allows us to state the basic problem in Invariant Ring Theory: Given a finite matrix group G , where every $g \in G$ is an $n \times n$ matrix, how do we find the finite generators of the invariant ring $\mathbb{C}[x_1, \dots, x_n]^G$?

2.2 The Fundamental Invariant Algorithm

Definition 2.2.1. A result of Theorem 2.1.11 is that every invariant $h \in \mathbb{C}[x_1, \dots, x_n]^G$ can be expressed in terms of the finite generators of $\mathbb{C}[x_1, \dots, x_n]^G$. These generators are generally referred to as a set of **fundamental invariants** for $\mathbb{C}[x_1, \dots, x_n]^G$. The set of fundamental invariants is often divided into two subsets, $\{\theta_1, \dots, \theta_n\}$, and $\{\eta_1, \dots, \eta_t\}$, where the first is called the set of **primary invariants**, and the second is called the set of **secondary invariants**. The primary invariants are distinguished from the secondary invariants in that the set of primary invariants consists of homogeneous invariant elements of positive degree that form what is called a **homogeneous system of parameters**. Most importantly, forming a homogeneous system of parameters requires the set of primary invariants to be algebraically independent. (For a more in-depth examination of the concept of a homogeneous system of parameters, see [5].) In practice, when trying to actually determine a set of fundamental invariants for a given group action, one gathers enough generators to form an algebraically independent set, and calls it the set of primary invariants. The secondary invariants are the

generators that have to be added to this set in order to make it a complete generating set for all of $\mathbb{C}[x_1, \dots, x_n]^G$.

In particular, every invariant $h \in \mathbb{C}[x_1, \dots, x_n]^G$ can be expressed as

$$h = \sum_{i=1}^t \eta_i \times p_i(\theta_1, \dots, \theta_n) \quad (2.2.1)$$

where p_1, \dots, p_n are suitable n -variate polynomials. In particular, the set $\{\theta_1, \dots, \theta_n, \eta_1, \dots, \eta_t\}$ forms a set of *fundamental invariants*, or generators, for $\mathbb{C}[x_1, \dots, x_n]^G$, where the θ_i are the *primary invariants* and the η_j are the *secondary invariants*. As we've seen with the symmetric polynomials, the set of fundamental invariants is not unique for a given group G , but given the degrees d_1, \dots, d_n of the primary invariants, the number of secondary invariants can be found by the formula

$$t = \frac{d_1 d_2 \dots d_n}{|G|} \quad (2.2.2)$$

Definition 2.2.2. For a given ideal I in a ring R , the **radical** of the ideal is denoted $Rad(I)$ and is defined as follows:

$$Rad(I) = \{r \in R \mid r^n \in I, \text{ for some } n \in \mathbb{Z}^+\}$$

Definition 2.2.3. Given a polynomial ring $\mathbb{C}[x_1, \dots, x_n]$, its **irrelevant ideal** is the ideal $M := (x_1, \dots, x_n)$, the ideal generated by the only the indeterminates.

Proposition 2.2.4. For the irrelevant ideal M , $M = (x_1, \dots, x_n) = \{p(x_1, \dots, x_n) \in \mathbb{C}[x_1, \dots, x_n] : p(0, \dots, 0) = 0 \text{ for some } p \in M\}$.

Proof. We have two things to prove: that every polynomial in $p \in M$ satisfies $p(0, \dots, 0) = 0$ and that any polynomial satisfying $p(0, \dots, 0) = 0$ is in M . \rightarrow Assume M is the irrelevant ideal. Then $p \in M$ is of the form $p = x_1 f_1 + x_2 f_2 + \dots + x_n f_n$ where $f_i \in \mathbb{C}[x_1, \dots, x_n]$. Then $p(0, \dots, 0) = 0 \cdot f_1(0, \dots, 0) + \dots + 0 \cdot f_n(0, \dots, 0) = 0$.

\leftarrow Assume $p(0, \dots, 0) = 0$. This is equivalent to saying that if you apply Algorithm 1.3.9 to p , with the divisors x_1, \dots, x_n , then the remainder is 0. Since the set is a Gröbner basis, we can apply Theorem 1.4.3, so $p \in M$. \square

Lemma 2.2.5. (*[13], p 53*) Let $G \subset GL(\mathbb{C}^n)$ be any finite matrix group, and let I_G denote the ideal in $\mathbb{C}[x_1, \dots, x_n]$ generated by all invariant homogeneous polynomials of positive degree. Then $\text{Rad}(I_G) = M$.

With this final lemma, we are ready to state the Fundamental Invariant Algorithm. (All following algorithms were taken from [13])

Algorithm 2.2.6. *Subalgorithm: Radical Containment*

1. Input: generators of an ideal I , where $I = \{b_1, \dots, b_m\}$, and the polynomial to be tested, f , where $b_1, \dots, b_m, f \in I$.
2. Let B be a Gröbner basis of $(b_1, \dots, b_m, fz - 1)$, where z is a new variable.
3. Let test = true if $1 \in B$, and false if it is not.
4. Output: test = true if $f \in \text{Rad}(I)$ and false if $f \notin \text{Rad}(I)$.

Algorithm 2.2.7. *Subalgorithm: Solvability of Homogeneous Equations*

1. Input: a set of homogeneous polynomials $f_1, \dots, f_m \in \mathbb{C}[x_1, \dots, x_n]$
2. Compute a Gröbner basis B of the ideal $I = (f_1, \dots, f_m)$.
3. Let test = true if a monomial of the form x_i^j occurs among the initial monomials in G for every i , for $1 \leq i \leq n$ and false if otherwise.
4. Output: If test = true, if there is no nonzero solution to the homogeneous system $\{f_1, \dots, f_m\}$, and if test = false, a nonzero solution does exist.

Algorithm 2.2.8. *Subalgorithm: Algebraic Dependence*

1. Input a set $F = \{f_1, \dots, f_m\} \subset \mathbb{C}[x_1, \dots, x_n]$.
2. Introduce "slack variables" $\{y_1, \dots, y_m\}$.
3. Compute a Gröbner basis B of $\{f_1 - y_1, f_2 - y_2, \dots, f_m - y_m\}$ with respect to the lexicographic ordering $x_1 > x_2 > \dots > x_n > y_1 > \dots > y_m$.
4. Let $B' = B \cap \mathbb{C}[y_1, \dots, y_m]$. That is, let B' be the set of all elements of B that contain no x_i variables.

5. Output: If $B' = \emptyset$, then F is algebraically independent. If $P(y_1, \dots, y_m) \in B'$, then $P(f_1, \dots, f_m) = 0$ is an algebraic dependence for the set $F \subset \mathbb{C}[x_1, \dots, x_n]$.

Algorithm 2.2.9. *Subalgorithm: Decomposition*

1. Input: Generators $\theta_1, \dots, \theta_n, \eta_1, \dots, \eta_t$, and a given polynomial $f \in \mathbb{C}[x_1, \dots, x_n]$ to be decomposed.
2. Introduce $n + t$ new "slack variables" $\{y_1, \dots, y_m, z_1, \dots, z_t\}$.
3. Compute a Gröbner basis B of $\{\theta_1 - y_1, \dots, \theta_n - y_n, \eta_1 - z_1, \dots, \eta_t - z_t\}$, with the respect to the monomial ordering defined as follows: the monomial term $Ax_1^{a_1} \dots x_n^{a_n} y_1^{c_1} \dots y_n^{c_n} z_1^{e_1} \dots z_t^{e_t}$ with multidegree $(a_1, \dots, a_n, c_1, \dots, c_n, e_1, \dots, e_t)$ has higher order than the monomial term $Bx_1^{b_1} \dots x_n^{b_n} y_1^{d_1} \dots y_n^{d_n} z_1^{f_1} \dots z_t^{f_t}$ with multidegree $(b_1, \dots, b_n, d_1, \dots, d_n, f_1, \dots, f_t)$ if the x -components of the two monomials satisfy $x_1^{a_1} \dots x_n^{a_n} > x_1^{b_1} \dots x_n^{b_n}$ with respect to the lexicographic ordering or else if the y -components satisfy $y_1^{c_1} \dots y_n^{c_n} > y_1^{d_1} \dots y_n^{d_n}$ with respect to the total degree ordering, or else if the z -component $z_1^{e_1}, \dots, z_t^{e_t} > z_1^{f_1}, \dots, z_t^{f_t}$ with respect to the lexicographic ordering.
4. Let $B' = B \cup \mathbb{C}[y_1, \dots, y_m, z_1, \dots, z_t]$; again, this represents all the polynomials in the Gröbner basis that have no x_i variables.
5. Apply Algorithm 1.3.9, with the elements of B acting as the b_1, \dots, b_m , and f as the polynomial to be divided. Set r_1 is the remainder produced by Algorithm 1.3.9. Set $Q := r_1$.
6. Apply Algorithm 1.3.9, with the elements of B' acting as the divisors, and Q as the polynomial to be divided. Set r_2 to be the remainder returned by the algorithm. Set $Q := r_2$.
7. Output: If $Q = 0$, then f can be expressed as in Equation 2.2.1. If $Q \neq 0$, then f cannot be expressed in that form.

Algorithm 2.2.10. Fundamental Invariant Algorithm

1. Input: Reynolds operator $*$ of a finite subgroup G of $GL(\mathbb{C}^n)$, and a fixed monomial order $m_1 < m_2 < \dots$ which refines the partial order given by the total degree on the set of monomials in $\mathbb{C}[x_1, \dots, x_n]$.

2. Let $t := 1$ and $\mathcal{Q} := \emptyset$.
3. Repeat $t := t + 1$ until m_t^* is not contained in $Rad(\mathcal{Q})$ (Go to Subalgorithm 2.2.6).
4. Let $\mathcal{Q} := \mathcal{Q} \cup \{m_t^*\}$. If $Rad(\mathcal{Q}) \neq M$ (Go to Subalgorithm 2.2.7) then go to Step 3.
5. If \mathcal{Q} has cardinality n
 - (a) then set $\mathcal{P} := \mathcal{Q}$.
 - (b) else modify \mathcal{Q} to an algebraically independent set \mathcal{P} of invariants with $Rad(\mathcal{P}) = M$ (go to Sub 2.2.8).
6. Write $\mathcal{P} = \{\theta_1, \dots, \theta_n\}$, let $S = \{1\}$, $t := 0$, and set $bound := \sum_{i=1}^n degree(\theta_i) - n$.
7. Let $t := t + 1$. If $degree(m_t) > bound$, then END. At this point \mathcal{P} and \mathcal{S} are the primary and secondary invariants respectively, and their union generates $\mathbb{C}[x_1, \dots, x_n]^G$ as a ring.
8. Test whether m_t^* can be expressed as in 2.2.1. (Go to Subalgorithm 2.2.9). If no, then $\mathcal{S} := \mathcal{S} \cup m_t^*$. Go to step 6.
9. Output: A set $\{\mathcal{P}, \mathcal{S}\}$ of fundamental invariants.

All of the above have been successfully written as programs in Maple, with two exceptions. First, in the event that \mathcal{Q} does not have cardinality n in Step 5, computing an algebraically independent set of primary invariants becomes very difficult, involving an algorithm (not shown here) that requires fractional exponents. Second, attempts to successfully program Maple to perform Gröbner bases with respect to the ordering defined in Subalgorithm 2.2.9 have so far proven impossible; while it is a fairly straightforward task by hand, the calculation is long and tedious. Both of these problems can be bypassed by restricting our choice of G to *reflection groups*.

Definition 2.2.11. A **reflection** can be loosely defined as inverting a geometric figure with respect to a line or a plane, although the formal generalized definition is a bit more technical. A typical example of a reflection is taking a point $(x_1, y_1) \in \mathbb{R}^2$ and switching them to (y_1, x_1) , which basically inverts the point with respect to the line $y = x$. A matrix or linear transformation

$\pi \in GL(\mathbb{C}^n)$ is called a *reflection* if precisely one *eigenvalue* is not equal to one (The eigenvalue of a matrix π being any element $\lambda \in \mathbb{C}$ such that $\pi\mathbf{v} = \lambda\mathbf{v}$ for some nonzero $\mathbf{v} \in \mathbb{C}^n$; computing eigenvalues is a common linear algebra procedure.) . A finite subgroup $G \subset GL(\mathbb{C}^n)$ is a **reflection group** if G is generated by reflections.

Polynomial rings invariant under reflection groups are of special interest because of the following theorem:

Theorem 2.2.12. (*Shephard-Todd-Chevalley Theorem*) ([13], p 44) *The invariant ring $\mathbb{C}[x_1, \dots, x_n]^G$ is generated by n algebraically independent homogeneous invariants iff G is a reflection group.*

What this means is that set of fundamental invariants for any polynomial ring $\mathbb{C}[x_1, \dots, x_n]^G$ is just $\{1, \theta_1, \dots, \theta_n\}$; there are no other secondary invariants, and Algorithm 2.2.10 produces only n primary invariants, bypassing both of the difficulties in programming the algorithm.

Example 2.2.13. Invariants under the group action S_2

Consider the polynomial ring $\mathbb{C}[x, y]$ being acted on by the symmetry group S_2 , as explained in Example 1.1.3. We want to find the generators for the invariant ring $\mathbb{C}[x, y]^{S_2}$. The matrix representation of S_2 is $\left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right\}$, with the latter element being the generator. Its eigenvalues are 1 and -1, so S_2 is a reflection group. With the matrix representation, we can explicitly define the Reynolds operator, as in Definition 2.1.10, so we can begin the algorithm.

- choose the lexicographic ordering $y > x$.
- Let f , our testing monomial, be the monomial x . Then $x^* = 1/2(x+y)$.
- x^* is definitely not in $Rad(\mathcal{Q})$, since the set is currently empty, so we set $\mathcal{Q} := \{x^*\}$.
- Applying Subalgorithm 2.2.7, the Gröbner basis of \mathcal{Q} is $\{x + y\}$, so $Rad(\mathcal{Q}) = (x) \neq M$. So we go back to step 3.
- Let $f = y$. Then $y^* = 1/2(x + y) = x^*$, which is certainly contained in $Rad(\mathcal{Q})$, since we already found that invariant.

- That concludes all the monomials with a total degree 1, so we start with any monomial of order two. Let $f = xy$. Then $xy^* = xy$.
- Running Subalgorithm 2.2.6 yields the Gröbner basis $\{zx^2 + 1, x + y\}$, which does not contain 1, so xy^* is not contained in $Rad(\mathcal{Q})$, and we set $\mathcal{Q} := \mathcal{Q} \cup \{xy^*\}$.
- Applying Subalgorithm 2.2.7, the Gröbner basis of \mathcal{Q} is $\{y + x, x^2 + 1\}$, so $Rad((\mathcal{Q})) = (x, y) = M$.
- The cardinality of \mathcal{Q} is 2, so we have a set of primary invariants. Since we are dealing with a reflection group, the only secondary invariant is 1. Thus, $\mathbb{C}[x, y]^{S_2} = (1, y + x, xy)$, which contains the set of symmetric polynomials for S_2 , which is exactly what we should get.

Note that this is not the only generating set for $\mathbb{C}[x, y]^{S_2}$. If we had used x^2 or y^2 , we would have wound up with the equivalent generator set $\mathbb{C}[x, y]^{S_2} = (1, y + x, x^2 + y^2)$, which is the subring generated by the power sums.

2.3 Algorithm for Determining Prime Ideals

Ideally, this section would be preceded by a few definitions, theorems, and, since it comes up, a whole bag of tricks to prove whether a polynomial is irreducible, and other matters concerning *fractions of rings*. However, it is of sufficient interest that removing it completely would be excessive. For now, it stands alone, to be better developed at a later point.

Let k be a field. Then the following algorithm can show whether an ideal $P \subset k[x_1, \dots, x_n]$ is a prime ideal.

Algorithm 2.3.1. (Prime Ideal Detection)

1. Input: the generators (f_1, \dots, f_t) for an ideal $P \subset k[x_1, \dots, x_n]$.
 2. Compute the reduced Gröbner basis $B = \{b_1, b_2, \dots, b_m\}$ with respect to the lexicographic monomial ordering $x_n > \dots > x_1$.
- (The elements of B lying in $k[x_1, \dots, x_i]$ form the reduced Gröbner basis $\{b_1, b_2, \dots, b_{m_i}\}$ for $P_i = P \cap k[x_1, \dots, x_i]$. (See Theorem 1.5.2))

3. Let $i := 1$. Set $P_i = P \cap k[x_1, \dots, x_i]$. Set $B_i := B \cap k[x_1, \dots, x_n]$. Let $i := i + 1$. Repeat while $i \leq n$.
 4. If $P_1 = 0$, go to Step 5. Else if the generator of P_1 is irreducible in $k[x_1]$, go to Step 5. Else set prime:="FALSE" and END.
- (For each $i \geq 2$ suppose P_{i-1} has been determined to be a prime ideal in $k[x_1, \dots, x_{i-1}]$ (otherwise, P is not a prime ideal in $k[x_1, \dots, x_n]$). Let $S = k[x_1, \dots, x_{i-1}]/P_{i-1}$ and let F be the fractional field of S . Continue the algorithm to determine whether P_i is a prime ideal in $k[x_1, \dots, x_i]$.)
5. let $i := i + 1$. If $i > n$, then set prime:="TRUE" and END.
 6. Set $S := k[x_1, \dots, x_{i-1}]/P_{i-1}$. Set $F := (S)$. (Where $()$ denote a field of fractions, not a generating set.)
 7. If $m_i = m_{i-1}$, then Go to Step 5. (P_i maps to the zero ideal in $S[x_i]$, and hence is prime). Else: (The image of P_i in $S[x_i]$ and in $F[x_i]$ is a nonzero ideal, and is generated by the images of $b_{m_{i-1}+1}, \dots, b_{m_i}$.)
 - (a) Let $t := 1$.
 - (b) Let $B_i' = B_i - B_{i-1}$, which is the set of elements of B_i that did not appear in B_{i-1} . Let t denote the subscript of the elements in B_i' .
 - (c) Apply the Euclidean Algorithm in $F[x_i]$ with b_t' as the polynomial to be divided, and the elements of the set B_{i-1} as the set to divide by. Note that fractional values are permitted. Set $b_t' := r$. Let $t := t + 1$. Repeat while $t < |B_i'|$. This gives us a generating set for the image of P_i in $F[x_i]$.
 - (d) Calculate the reduced Gröbner basis of (B_i') . (This will result in a single polynomial, $im(h)$)
 - (e) If h of $im(h)$ is irreducible in $F[x_i]$, go to Step 8. Else prime:="false"; END.

(Note that after applying the Euclidean algorithm to the generators of the image of P_i in $F[x_i]$ we can multiply by a single element of S to "clear denominators" in each equation so that all remainders (and in particular the last nonzero remainder $h(x_i)$) will be elements in the image of P_i .)

8. Let $a \in k[x_1, \dots, x_{i-1}]$ be the leading coefficient of $h(x_i)$ (as a polynomial in x_i).
 9. Compute the reduced Gröbner basis C in $k[x_1, \dots, x_{i-1}, t]$ for the ideal generated by P_i and $1 - at$ with respect to the lexicographic monomial ordering $t > x_i > \dots > x_1$.
 10. If $C \cap k[x_1, \dots, x_i] = B_i$, then go to Step 3). Otherwise, set prime:=”FALSE” and END.
- (If all P_i for $i = 1, \dots, n$ are prime ideals, then P is a prime ideal in $k[x_1, \dots, x_n]$.)

Example 2.3.2. Examples Using Algorithm

Example 1: Determine whether $I = (y^3 - xz, xy^2 - z^2)$ in $\mathbb{Q}[x, y, z]$ is a prime ideal.

- Calculate the Gröbner basis for I . The reduced Gröbner basis for I with respect to the ordering $x > y > z$ is $B = \{y^5 - z^3, xz - y^3, xy^2 - z^2\}$.
- Find all $I_i B_i, i = 1, 2, 3$, where $I_i = I \cap \mathbb{Q}[x, y, z]$, B_i is the reduced Gröbner basis of I_i .
 $I_1 = I \cap \mathbb{Q}[z] = (0), B_1 = \{\emptyset\},$
 $I_2 = I \cap \mathbb{Q}[y, z] = (y^5 - z^3), B_2 = \{y^5 - z^3\}$ and
 $I_3 = I \cap \mathbb{Q}[x, y, z] = I, B_3 = B.$
- $I_1 = (0)$, and is thus prime, so we move to I_2 .
- $I_2 = (y^5 - z^3)$, and we must set up our respective S and F . $S = \mathbb{Q}[z]/0 \cong \mathbb{Q}[z], F = \mathbb{Q}(z)/0 \cong \mathbb{Q}(z).$
- The number of elements in B_2 is greater than the number of elements in B_1 , and we set $B_2' = \{y^4 - z^3\}$.
- Since $B_1 = \{\emptyset\}$, reducing the elements of B_2' by the elements of B_1 is easy; the set remains the same, except we apply an overline to z to emphasize that the element has been calculated in $F[y]$: $y^5 - \bar{z}^3 = im(h(y)).$
- Next, we check to see whether $y^5 - z^3$ is irreducible in $F[y]$, which it is (according to Maple), and we move on to the next step.

- $a = LC(y^5 - z^3) = 1$.
- C is the Gröbner basis of $(1-t, y^5 - z^3)$ with respect to the lexicographic monomial ordering $t > y > z$. $C = \{y^5 - z^3, 1 - t\}$.
- $(1 - t, y^5 - z^3) \cap S = I_2$, so I_2 is a prime ideal. We move on to I_3 .
- We set up our S and F : $S = \mathbb{Q}[y, z]/(y^5 - z^3)$, $F = \mathbb{Q}(y, z)/(y^5 - z^3)$.
- $B_3' = \{xz - y^3, xy^2 - z^2\}$.
- Apply the Euclidean Algorithm to $xz - y^3$, with $y^5 - z^3$ as the divisor.
- Determine whether I_1 is a prime ideal in $\mathbb{Q}[z]$: $I_1 = (0)$, so it is a prime ideal in $\mathbb{Q}[z]$.
- Set $S = \mathbb{Q}[z]/(0)$, and $F = \mathbb{Q}(z)/(0)$. The next step is to apply the Euclidean algorithm to find the greatest common divisor of the images of the generators of I_2 in $F[y]$. Since we only have one element in this set of generators, this greatest common divisor is just $h(y) = y^5 - \bar{z}^3$. This polynomial is irreducible in $\mathbb{Q}(z)[y]/I_1$, and hence we don't yet know if I_2 is prime or not, and we move on to the next step.
- Calculate the reduced Gröbner basis in $\mathbb{Q}[t, y, z]$ of $(y^5 - z^3, 1 - t)$ with respect to the lexicographic monomial ordering $t > y > z$. The basis is: $\{y^5 - z^3, 1 - t\}$. The intersection of this basis with the basis of I_2 gives $\{y^5 - z^3\}$, so I_2 is a prime ideal.
- Apply the same method to I. We set $S = \mathbb{Q}[y, z]/I_2$ and $F = \mathbb{Q}(y, z)/I_2$. The image of I in $S[x]$ is generated by the elements $\{\bar{y}^5 - \bar{z}^3 = 0, x\bar{z} - \bar{y}^3, x\bar{y}^2 - \bar{z}^2\}$. The greatest common divisor in $F[x]$ of $x\bar{z} - \bar{y}^3$ and $x\bar{y}^2 - \bar{z}^2$ is \bar{z}^2 , which is reducible in $F[x]$ as $\bar{z} * \bar{z}$, which means I is not a prime ideal in the field $\mathbb{Q}[x, y, z]$.

Example 2: Determine whether $I = (x^2 - yz, w^2 - x^4z)$ is a prime ideal in $\mathbb{Q}[x, y, z, w]$.

In this case, the reduced Gröbner basis is $\{x^2 - yz, z^3y^2 - w^2\}$. Then P_1 and P_2 are both (0) , so we can skip to $P_3 = (z^3y^2 - w^2)$. Set $S = \mathbb{Q}[z, w]$ and $R = \mathbb{Q}(z, w)$. The image of P_3 in $F[y]$ is generated by $h(y) = z^3y^2 - w^2$, which is irreducible in R . We set $a = z^3$ and the reduced Gröbner basis for

the set $\{z^3y^2 - w^2, 1 - z^3t\}$ is $B_3 = \{z^3y^2 - w^2, y^2 - tw^2, z^3t - 1\}$. The intersection of the ideal generated by B_2 and I_3 is I_3 , so it is an ideal.

Last, we consider I_4 . Let $S = \mathbb{Q}[y, z, w]$ and $R = \mathbb{Q}(y, z, w)$. Then the greatest common divisor in $F[x]$ of $b_1 = x^2 - \bar{y}z$ and $b_2 = \bar{z}^3\bar{y}^2 - \bar{w}^2 = 0$ is just $x^2 - \bar{y}z$, which is irreducible in $F[x]$. We set $a = 1$ and calculate the reduced Gröbner basis for the set $\{x^2 - yz, z^3y^2 - w^2, 1 - t\}$, which is $B_4 = \{x^2 - yz, 1 - t\}$. The intersection of the ideal generated by B_4 and I is again I , so I is a prime ideal in $\mathbb{Q}[x, y, z, w]$.

Theorem 2.3.3. Justification of Algorithm

The basic idea behind the algorithm is that we can use the fact that $k[x_1, \dots, x_i] = k[x_1, \dots, x_{i-1}][x_i]$ to inductively determine whether the ideals $P_i \cap k[x_i, \dots, x_n]$ are prime. In general, suppose that R is a commutative ring. If P is a prime ideal in $R[x]$, then $P \cap R$ is a prime ideal in R , since P is just elements of $P \cap R$ with the indeterminate x . Thus, since $P \cap R$ is a prime ideal, $S = R/(P \cap R)$ is an integral domain. Let F denote the quotient field of S . We have two natural ring homomorphisms, from $R[x]$ into $S[x]$, and from $S[x]$ into $F[x]$.

Statement 2.3.4. Suppose R is a commutative ring with 1 and I is an ideal in $R[x]$. Then I is a prime ideal in $R[x]$ iff

1. $J = I \cap R$ is a prime ideal in R ; which is equivalent to the condition that $S = R/J$ is an integral domain and
2. If \bar{I} is the image of I in $S[x]$ then $\bar{I}F[x]$ is a prime ideal in $F[x]$ satisfying $\bar{I}F[x] \cap S[x] = \bar{I}$.

Proof. Suppose I is a prime ideal in $R[x]$, so that $J = I \cap R$ is a prime ideal in R and $S = R/J$ is an integral domain. The kernel of the homomorphism from $R[x] \mapsto S[x] = (R/J)[x]$ is $J[x]$, which is contained in $I[x]$, so by the First Isomorphism Theorem, we have $R[x]/I \cong S[x]/\bar{I}$. Since $R[x]/I$ is an integral domain, $\bar{I} \cap S$ are the images of the elements in $R \cap I$, so $\bar{I} \cap S = 0$. Since the ring $F[x]$ is the localization of $S[x]$ with respect to the multiplicatively closed set $S - \{0\}$, condition ii) follows by the fact that there exists a bijective mapping between the prime ideals of S where $\bar{I} \cap S = 0$ and the prime ideals of F .

Conversely, if I is not prime, then either J is not prime in R or J is prime in R , but \bar{I} is not prime in $S[x]$. In the latter case, either $\bar{I}F[x]$ is not prime in $F[x]$ or, by the property that the prime ideals of an integral domain

and the prime ideals of its localization have a bijective correspondence, $\bar{I} \neq \bar{I}F[x] \cap S[x]$. Thus, either 1 or 2 fails, completing the proof. \square

Since $F[x]$ is a Euclidean Domain, the ideal $\bar{I}F[x]$ is principal, and is prime iff $h(x)$ is either (0) or is irreducible in $F[x]$. (We know this from basic results concerning Euclidean Domains.) Suppose $h(x)$ is an element in I whose image in $S[x]$ has leading coefficient $a \in S$. a can be used as follows to give a bound on the denominators necessary for $\bar{I}F[x] \cap S[x] = \bar{I}$.

Statement 2.3.5. Let S be an integral domain with fraction field F and let A be a nonzero ideal in $S[x]$. Suppose $AF[x] = (h(x))$ where $h(x)$ is a polynomial in $S[x]$ with leading coefficient $a \in S$. Let S_a be the localization of S with respect to the powers of a . Then

1. $AF[x] \cap S[x] = AS_a[x] \cap S[x]$, and
2. If \mathcal{A} denotes the ideal generated by A and $1 - at$ in the polynomial ring $S[x, t]$, then $\mathcal{A}S_a[x] \cap S[x] = \mathcal{A} \cap S[x]$.

Proof. Since $S_a \subseteq F$, the containment $AS_a[x] \subseteq AF[x] \cap S_a[x]$ follows automatically. Suppose now that $f(x) \in AF[x] \cap S_a[x]$. If the leading term of $f(x)$ is sx^N and the leading term of $h(x)$ is ax^m , then since $AF[x] = (h(x))$ we have $N \geq m$. Then the polynomial $f(x) - (s/a)x^{N-m}h(x)$ is again in $AF[x] \cap S_a[x]$ and is of lower degree than $f(x)$. Iterating, we see that $f(x)$ can be written as a polynomial in $S_a[x]$ times $h(x)$, so $f(x) \in AS_a[x]$. Intersecting both sides of $AF[x] \cap S_a[x] = AS_a[x]$ with $S[x]$ gives the first statement.

To prove the second statement, suppose first that $f(x) \in \mathcal{A} \cap S[x]$. Then we can write $f(x) = f_1(x, t)b(x) + f_2(x, t)(1 - at)$ for some polynomials $b(x) \in \mathcal{A}$ and $f_1, f_2 \in S[x, t]$. Substituting $t = 1/a$ gives $f(x) = f_1(x, 1/a)b(x)$, and since $f_1(x, 1/a) \in S_a[x]$, we obtain $f(x) \in AS_a[x] \cap S[x]$. Conversely, suppose that $f(x) = b(x)g(x) \in S[x]$ where $g(x) \in S_a[x]$ and $b(x) \in A$. If a^N is the largest power of a appearing in the denominators of the coefficients of $g(x)$ then $a^N g(x) \in S[x]$. We can then express $f(x)$ as $f(x) = (at)^N f(x) + (1 - (at)^N)f(x) = b(x)t^N(a^N g(x)) + (1 - (at)^N)f(x)$ which shows that $f(x) \in S[x]$, giving the reverse containment and completing the proof. \square

Proof. Final Proof Note how both of these statements connect with the actual process of the algorithm. The multivariable equivalent to Statement 2.3.4 would be something like: I is a prime ideal in $R[x_1, \dots, x_n]$ under the lexicographic monomial ordering $x_n > \dots > x_1$ if and only if I_{n-1} is a prime

ideal in $R[x_1, \dots, x_n]$ and the image of I , denoted \bar{I} in $S[x_n] = R[x_1, \dots, x_{n-1}]$ is such that $\bar{I}F[x_n]$ is a prime ideal in $F[x_n]$ satisfying $\bar{I}F[x_n] \cap S[x_n] = \bar{I}$.

The first part of the statement is basically the inductive step of the algorithm, where we first prove that the I_1, \dots, I_{n-2} elimination ideals are prime before moving on to the next one. The rest of the algorithm is given by the second part of the statement, although to really demonstrate that we need a slight rewording for Statement 2.3.5: (This Statement was a little thicker in terminology, so please bear with)

Let \bar{I} be a nonzero ideal in $S[x]$, where $S = R[x_1, \dots, x_{n-1}]$ I_{n-1} and F is the fractional field of S . Given that $F[x_n]$ is a Euclidean Domain, the ideal $\bar{I}F[x_n]$ is a prime ideal iff it is generated by a single polynomial term $h(x_n) \in S[x_n]$, where $h(x_n)$ is either 0 or irreducible in $F[x_n]$. Furthermore, let a and S_a be as in the original Statement 2.3.5. Then $\bar{I}F[x_n] \cap S[x_n] = \bar{I}S_a[x_n] \cap S[x_n] = (b_{1,n-1}, \dots, b_{m,n-1}, 1 - at) \cap S[x_n] = I_{n-1}$, where $b_{1,n-1}, \dots, b_{m,n-1}$ is the reduced Gröbner basis for I_{n-1} .

Everything prior the sentence beginning "Furthermore" gives the conditions necessary to fulfill the first part of the second item in Statement 2.3.4, that $\bar{I}F[x]$ is a prime ideal in $F[x]$, and corresponds to step 7 in the algorithm. The rest of the restatement gives the necessary requirement to show that $\bar{I}F[x] \cap S[x] = \bar{I}$, which is that $I_i = (I_i + (1 - at)) \cap S[x]$, which gives the final Steps 8 through 10 of the Algorithm.

We complete the section with a more formal general validation:

Let P be a prime ideal in $k[x_1, \dots, x_n]$, and let P_i for $i = 1, \dots, n$ be the intersection of P with $k[x_1, \dots, x_i]$. We use the preceding propositions to prove inductively that the ideals $P_1, \dots, P_n = P$ are prime.

The ideal P_i will be prime in the Euclidean domain $k[x_1]$ iff it is 0 or generated by an irreducible polynomial. Suppose that $i \geq 2$ and we have already proven that P_{i-1} is a prime ideal in $k[x_1, \dots, x_{i-1}]$, so that $S = k[x_1, \dots, x_{i-1}]/P_{i-1}$ is an integral domain. If F denotes the quotient field of S , then by the first Proposition, P_i is a prime ideal in $k[x_1, \dots, x_i]$ iff its image in $(k[x_1, \dots, x_{i-1}]/P_{i-1})[x_i] = S[x_i]$ is a saturated ideal whose extension to the Euclidean Domain $F[x_i]$ is a prime ideal. Suppose $h(x_i) \in S[x_i]$ is a generator for this ideal and a is the leading coefficient of $h(x_i)$. Then $(h(x_i))$ is a prime ideal in $F[x_i]$ iff $h(x_i) = 0$ or $h(x_i)$ is an irreducible polynomial. By the second Proposition, the image of P_i in $S[x_i]$ will be saturated iff it equals $\mathcal{A} \in S[x_i]$ where \mathcal{A} is the ideal generated by P_i and $1 - at$ in $S[x_i, t]$. This latter condition can be checked in $k[x_i, \dots, x_i, t]$; it is the equivalent to checking that the intersection generated by P_i and $1 - at$ in $k[x_1, \dots, x_i, t]$ with

$k[x_1, \dots, x_i]$ is just P_i . □

Combining these results with the basic calculations of the Gröbner basis gives the algorithm listed in the first section.

2.4 Torus Invariants and Integer Programming

Define a set $\mathcal{M}_{\mathcal{A}} := \{(v_1, \dots, v_n) \in \mathbb{Z}^n - 0 : v_1, \dots, v_n \geq 0 \text{ and } \mathbf{v} \cdot \mathcal{A} = 0\}$. Given the integer $n \times d$ -matrix \mathcal{A} we associate a group of diagonal $n \times n$ -matrices:

$$\Gamma_{\mathcal{A}} = \left\{ \text{diag} \left(\prod_{i=1}^d t_i^{a_{1i}}, \prod_{i=1}^d t_i^{a_{2i}}, \dots, \prod_{i=1}^d t_i^{a_{ni}} : t_1, \dots, t_d \in \mathbb{C}^x \right) \right\}$$

or $\Gamma_{\mathcal{A}} = \left(\begin{array}{cccc} t_1^{a_{11}} t_2^{a_{12}} \dots t_d^{a_{1d}} & 0 & \dots & 0 \\ 0 & t_1^{a_{21}} t_2^{a_{22}} \dots t_d^{a_{2d}} & \dots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & t_1^{a_{d1}} t_2^{a_{d2}} \dots t_d^{a_{dd}} \end{array} \right)$ This

matrix group $\Gamma_{\mathcal{A}}$ is isomorphic to the group $(\mathbb{C}^x)^d$ of invertible diagonal $d \times d$ -matrices, which is called the *the d -dimensional algebraic torus*. We call $\Gamma_{\mathcal{A}}$ the torus defined by \mathcal{A} . There is a connection between the invariant ring $\mathbb{C}[x_1, \dots, x_n]^{\Gamma_{\mathcal{A}}}$ and the set $\mathcal{M}_{\mathcal{A}}$:

Proposition 2.4.1. (a) A monomial $\mathbf{x}^v = x_1^{v_1} \dots x_n^{v_n}$ is $\Gamma_{\mathcal{A}}$ -invariant iff $v = (v_1, \dots, v_n) \in \mathcal{M}_{\mathcal{A}}$.

(b) A finite set $\mathcal{H} \subset \mathbb{Z}^n$ equals the Hilbert basis of $\mathcal{M}_{\mathcal{A}}$ iff the invariant ring $\mathbb{C}[x_1, \dots, x_n]^{\Gamma_{\mathcal{A}}}$ is minimally generated as a \mathbb{C} -algebra by $\{\mathbf{x}^v \in \mathcal{H}\}$.

Proof. The image of \mathbf{x}^v under a torus element of $\Gamma_{\mathcal{A}}$ equals:

$$(x_1 \prod_{i=1}^d t_i^{a_{1i}})^{v_1} \dots (x_n \prod_{i=1}^d t_i^{a_{ni}})^{v_n} = (x_1^{v_1} \dots x_n^{v_n}) \cdot \prod_{i=1}^d t_i^{\sum_{j=1}^n v_j a_{ji}}.$$

Therefore, \mathbf{x}^v is invariant under the action of $\Gamma_{\mathcal{A}}$ iff $\sum_{j=1}^n v_j a_{ji} = 0$, for $i = 1, \dots, d$. This is the same as saying that $v \cdot \mathcal{A} = 0$, which is how we defined $\mathcal{M}_{\mathcal{A}}$.

As for part b), the Hilbert basis \mathcal{H} of $\mathcal{M}_{\mathcal{A}}$ is defined as the minimal subset for which every $\beta \in \mathcal{M}_{\mathcal{A}}$ can be expressed as

$$\beta = \sum_{v \in \mathcal{H}} c_v \cdot v,$$

where the c_v are non-negative integers. This expression, applied to the correspondent invariant action, becomes $\mathbf{x}^v = \prod_{v \in \mathcal{H}} (\mathbf{x}^v)^{c_v}$, which says every element in $\Gamma_{\mathcal{A}}$ is generated by some product of $\{\mathbf{x}^v : v \in \mathcal{H}\}$. \square

What this means, essentially, is that since we know how to use Gröbner bases to determine the minimal generating set of an invariant polynomial ring and how to use them to determine whether an element is in an invariant ring, we can apply algorithms that use Gröbner bases to determine the Hilbert basis of the set $\mathcal{M}_{\mathcal{A}}$ and whether it is nonempty or not.

Algorithm 2.4.2. Algorithm for Non-empty Set Determination This algorithm determines whether the set $\mathcal{M}_{\mathcal{A}}$, as previously defined, is empty or not for a given $n \times d$ -matrix \mathcal{A} . If the set is nonempty, it also gives an element of the set.

1. Begin with a $n \times d$ -matrix \mathcal{A} . Compute any reduced Gröbner basis B for the kernel of the \mathbb{C} -algebra homomorphism

$$\mathbb{C}[x_1, x_2, \dots, x_n] \rightarrow \mathbb{C}[t_1, \dots, t_d, t_1^{-1}, \dots, t_d^{-1}], x_i \mapsto \prod_{j=1}^d t_j^{a_{ij}}.$$

Since the kernel is set of elements in $\mathbb{C}[x_1, x_n]$ that get mapped to 0, this is equivalent to calculating the reduced Gröbner basis for the set of elements generated by all $x_i - \prod_{j=1}^d t_j^{a_{ij}}$.

2. If B contains an element of the form $x_1^{\beta_1} x_2^{\beta_2} \dots x_n^{\beta_n} - 1$, then $\mathcal{M}_{\mathcal{A}}$ is nonempty, and $(\beta_1, \dots, \beta_n) \in \mathcal{M}_{\mathcal{A}}$. If it does not contain an element of that form, then $\mathcal{M}_{\mathcal{A}}$ is empty.

One further note. In step 1, we may encounter negative exponents a_{ij} . To deal with these, introduce the new variable t_0 , and choose any elimination order $\{t_0, \dots, t_d\} \succ \{x_1, \dots, x_n\}$. Using the relation $t_0 t_1 \dots t_d = 1$, clear the denominators in $x_i - \prod_{j=1}^d t_j^{a_{ij}}$, for $i = 1, \dots, n$. For the resulting $n + 1$ variables, compute a Gröbner basis B' with respect to \prec . Let $B = B' \cap \mathbb{C}[x_1, \dots, x_n]$.

Proof. Let I denote the kernel of the map in the algorithm. This is a prime ideal in the polynomial ring $\mathbb{C}[x_1, \dots, x_n]$, having the generic point $(\prod_{i=1}^d t_i^{a_{1i}}, \dots, \prod_{i=1}^d t_i^{a_{ni}})$. By the proof of Theorem 2.4.1, a monomial \mathbf{x}^β is invariant under $\Gamma_{\mathcal{A}}$ iff \mathbf{x}^β is congruent to 1 modulo I , that is, it is 1 after being reduced by the generators of I . Therefore, if the algorithm outputs a vector β , then β must lie in $\mathcal{M}_{\mathcal{A}}$. What remains to be shown is that the algorithm fails if and only if $\mathcal{M}_{\mathcal{A}} = \emptyset$. We prove this by contradiction. Suppose the algorithm fails and $\mathcal{M}_{\mathcal{A}} \neq \emptyset$ and let $\beta \in \mathcal{M}_{\mathcal{A}}$. Then $\mathbf{x}^\beta - 1$ lies in the ideal I , and hence the normal form of \mathbf{x}^β modulo the Gröbner basis B equals 1. In each step in the reduction of \mathbf{x}^β , a monomial is reduced to another monomial. In the last step, some monomial \mathbf{x}^γ reduces to 1. This implies $\mathbf{x}^\gamma - 1 \in B$. This contradicts the assumption that the algorithm fails. Thus, the algorithm fails iff $\mathcal{M}_{\mathcal{A}} = \emptyset$. \square

Algorithm 2.4.3. Algorithm for Finding the Hilbert Basis

1. Take the variables $t_1, \dots, t_d, x_1, \dots, x_n, y_1, \dots, y_n$. Fix any elimination monomial order $\{t_1, \dots, t_d\} \succ \{x_1, \dots, x_n\} \succ \{y_1, \dots, y_n\}$. Let $J_{\mathcal{A}}$ denote the kernel of the \mathbb{C} -algebra homomorphism

$$\mathbb{C}[x_1, \dots, x_n, y_1, \dots, y_n] \rightarrow \mathbb{C}[t_1, \dots, t_d, t_1^{-1}, \dots, t_d^{-1}, y_1, \dots, y_n], x_i \mapsto y_i \prod_{j=1}^d t_j^{a_{ij}}, y_i \mapsto y_i$$

.

In other words, $J_{\mathcal{A}} = (x_i - y_i \prod_{j=1}^d t_j^{a_{ij}})$ for all $i = 1, \dots, n$.

2. Compute the reduced Gröbner basis B with respect to \prec for the ideal $J_{\mathcal{A}}$.
3. The Hilbert basis \mathcal{H} of $\mathcal{M}_{\mathcal{A}}$ consists of all vectors β such that $\mathbf{x}^\beta - \mathbf{y}^\beta$ is in B .

Proof. Note that $J_{\mathcal{A}}$ is a homogeneous prime ideal and there is no monomial contained in $J_{\mathcal{A}}$. By the same reasoning as the previous algorithm, a vector $\beta \in \mathbb{N}^n$ lies in $\mathcal{M}_{\mathcal{A}}$ iff the monomial difference $\mathbf{x}^\beta - \mathbf{y}^\beta$ lies in $J_{\mathcal{A}}$. We wish to show that the finite subset $\mathcal{H} \subset \mathcal{M}_{\mathcal{A}}$ created by the algorithm spans the monoid $\mathcal{M}_{\mathcal{A}}$. Suppose that \mathcal{H} does not span $\mathcal{M}_{\mathcal{A}}$. Then there exists a minimal (with respect to divisibility) monomial \mathbf{x}^β such that $\beta \in \mathcal{M}_{\mathcal{A}}$, but β is not a sum of elements in \mathcal{H} . The polynomial $\mathbf{x}^\beta - \mathbf{y}^\beta$ lies in $J_{\mathcal{A}}$, so it reduces to zero under division by the elements of B . By the choice of

monomial order, the first reduction step replaces \mathbf{x}^β by some monomial $\mathbf{x}^\gamma \mathbf{y}^\delta$, where $\delta = \beta - \gamma \neq 0$. Therefore

$$\mathbf{x}^\gamma \mathbf{y}^\delta - \mathbf{y}^\beta = \mathbf{y}^\delta (\mathbf{x}^\gamma - \mathbf{y}^\gamma) \in J_{\mathcal{A}}$$

Since $J_{\mathcal{A}}$ is a prime ideal, not containing any monomials, we conclude that $\mathbf{x}^\gamma - \mathbf{y}^\gamma$ lies in $J_{\mathcal{A}}$. This implies that γ lies in $\mathcal{M}_{\mathcal{A}}$. By our minimality assumption on β , we have that both γ and δ can be written as sums of elements in \mathcal{H} . Therefore $\beta = \gamma + \delta$ can be written as sums of elements in \mathcal{H} . This is a contradiction, and the proof is complete. \square

Example 2.4.4. Examples using the Algorithms

Note: In both the following examples, \mathcal{A} is chosen to be a 4×1 -matrix, to simplify calculations. Since there is only one t -variable, we replace t_1 with just t .

1. Let $\mathcal{A} = (3, 1, -2, -2)^T$. Determine whether the set $\mathcal{M}_{\mathcal{A}}$ is nonempty, and if it is, determine its Hilbert basis.

Applying Algorithm 2.4.2, we first get the set $\{x_1 - t^3, x_2 - t^1, x_3 - t^{-2}, x_4 - t^{-2}\}$. Taking the first term with a negative exponent, $x_3 - t^{-2} = t_0^2 t^2 x_3 - t_0^2 t^2 t^{-2} = t_0^2 t^2 x - t_0^2$, and we are left with the set $\{x_1 - t^3, x_2 - t, t^2 x_3 - 1, t^2 x_4 - 1\}$. Calculating the Gröbner basis of this set gives $B' = \{x_3 - x_4, x_4 x_2^2 - 1, x_1 - x_2^3, t - x_2\}$, so $B = \{x_3 - x_4, x_4 x_2^2 - 1, x_1 - x_2^3\}$. $x_4 x_2^2 - 1$ is of the proper form, so the set is non-empty, and $(0, 2, 0, 1) \in \mathcal{M}_{\mathcal{A}}$.

For the second part, since we've already shown how the denominator-clearing step works, we'll skip straight to the set the Gröbner algorithm is applied to: $\{x_1 - t^3 y_1, x_2 - t y_2, t^2 x_3 - y_3, t^2 x_4 - y_4\}$. The reduced Gröbner basis with respect to the lexicographic monomial order $t \succ x_1 \succ \dots \succ x_4 \succ y_1 \succ \dots \succ y_4$ equals $B = \{t^3 y_1 - x_1, t^2 x_2 y_1 - x_1 y_2, t^2 x_3 - y_3, t^2 x_4 - y_4, t x_1 x_3^2 - y_1 y_3^2, t x_1 x_3 x_4 - y_1 y_3 y_4, t x_1 x_4^2 - y_1 y_4^2, t x_2^2 y_1 - x_1 y_2^2, t x_2 x_3 - y_2 y_3, t x_2 x_4 - y_2 y_4, t y_1 y_3 - x_1 x_3, t y_1 y_4 - x_1 x_4, t y_2 - x_2, \underline{x_1^2 x_3^3 - y_1^2 y_3^3}, \underline{x_1^2 x_3^3 x_4} - y_1^2 y_3^3 y_4, \underline{x_1 x_2 x_4^2} - y_1^2 y_3 y_4^2, \underline{x_1^2 x_4^3} - y_1^2 y_4^3, \underline{x_1 x_2 x_3^2} - y_1 y_2 y_3^2, \underline{x_1 x_2 x_3 x_4} - y_1 y_2 y_3 y_4, \underline{x_1 x_2 x_4^2} - y_1 y_2 y_4^2, x_1 x_3 y_2 - x_2 y_1 y_3, x_1 x_4 y_2 - x_2 y_1 y_4, x_1 y_2^3 - x_2^3 y_1, \underline{x_2^2 x_3} - y_2^2 y_3, \underline{x_2^2 x_4} - y_2^2 y_4, x_3 y_4 - x_4 y_3\}$ and the Hilbert basis of $\mathcal{M}_{\mathcal{A}}$ consists of the nine underlined polynomials: $\mathcal{H} =$

$\{(2, 0, 3, 0), (2, 0, 2, 1), (2, 0, 1, 2), (2, 0, 0, 3), (1, 1, 2, 0), (1, 1, 1, 1), (1, 1, 0, 2), (0, 2, 0, 1), (0, 2, 1, 0)\}$.

2. Let's try the same thing, but this time with $\mathcal{A} = (4, 1, -2, -3)^T$. For the non-empty determination, we have the starting set $\{x_1 - t^4, x_2 - t, t^2x_3 - 1, t^3x_4 - 1\}$. The reduced Gröbner basis is $B = \{t - x_2, x_1 - x_2^4, \underline{x_2^2x_3 - 1}, x_2x_3^2 - x_4, x_2x_4 - x_3, x_3^3 - x_4^2\}$, and $(0, 2, 1, 0) \in \mathcal{M}_{\mathcal{A}}$. As for the spanning set, we begin with the set $\{x_1 - t^4y_1, x_2 - ty_2, t^2x_3 - y_3, t^3x_4 - y_4\}$, the generator for $J_{\mathcal{A}}$. Its reduced Gröbner basis is $B = \{t^4y_1 - x_1, t^3x_2y_1 - y_2x_1, t^3x_4 - y_4, t^2x_1x_4^2 - y_1y_4^2, t^2x_2^2y_1 - x_1y_2^2, t^2x_2x_4 - y_2y_4, t^2x_3 - y_3, t^2y_1y_3 - x_3x_1, \underline{tx_1^2x_4^3 - y_1^2y_4^3}, \underline{tx_1x_2x_4^2 - y_1y_2y_4^2}, \underline{tx_1x_3x_4 - y_1y_3y_4}, \underline{tx_2^3y_1 - x_1y_2^3}, \underline{tx_2^2x_4 - y_2^2y_4}, \underline{tx_2x_3 - y_2y_3}, \underline{tx_2y_1y_3 - x_1x_4y_2}, \underline{tx_3^2y_4 - x_4y_3^2}, \underline{tx_4y_3 - x_3y_4}, \underline{ty_1y_4 - x_1x_4}, \underline{ty_2 - x_2}, \underline{x_1^3x_4^4 - y_1^3y_4^4}, \underline{x_1^2x_2x_4^3 - y_1^2y_2y_4^3}, \underline{x_1^2x_3x_4^2 - y_1^2y_3y_4^2}, \underline{x_1x_2^2x_4^2 - y_1y_2^2y_4^2}, \underline{x_1x_2x_3x_4 - y_1y_2y_3y_4}, \underline{x_1x_3^2 - y_1y_3^2}, \underline{x_1x_3y_2^2 - x_2^2y_1y_3}, \underline{x_1x_4^2y_3 - x_3y_1y_4^2}, \underline{x_1x_4y_2 - x_2y_1y_4}, \underline{x_1y_2^4 - x_2^4y_1}, \underline{x_2^3x_4 - y_2^3y_4}, \underline{x_2^2x_3 - y_2^2y_3}, \underline{x_2x_3^2y_4 - x_4y_2y_3^2}, \underline{x_2x_4y_3 - x_3y_2y_4}, \underline{x_3^3y_4^2 - x_4^2y_3^3}\}$. Taking the eight underlined polynomials, the Hilbert basis is $\mathcal{H} = \{(3, 0, 0, 4), (2, 1, 0, 3), (2, 0, 1, 2), (1, 2, 0, 2), (1, 1, 1, 1), (1, 0, 2, 0), (0, 3, 0, 1), (0, 2, 1, 0)\}$. This is almost a basis corresponding to orthogonal complement of \mathcal{A} ; but not quite, since we used a different definition of spanning set in order to get the Gröbner algorithm to work for the variables. We can, however, obtain the true basis fairly easily through linear algebra.

Chapter 3

Algebraic Geometry

Sadly, we didn't quite get to the point where we could apply our studies in invariant theory to algebraic geometry, but we did discover many useful algebraic tools, and in this chapter, we will briefly go over them.

3.1 Noetherian Rings

Before we dive into the geometric aspect, we review some basic concepts of *Noetherian* rings. Recall Definition 1.2.1. The following definition is equivalent:

Definition 3.1.1. A commutative ring R is said to be *Noetherian* or to satisfy the ascending chain condition on ideals (abbreviated as the ACC on ideals) if there is no infinitely increasing chain of ideals in R ; whenever $I_1 \subseteq I_2 \subseteq \dots$ is an increasing chain of ideals in R , there is a positive integer m such that $I_k = I_m$ for all $k > m$, $k, m \in \mathbb{N}$.

Corollary 3.1.2. *If I is any ideal of the Noetherian ring R , then the quotient ring R/I is a Noetherian ring. Any homomorphic image of a Noetherian ring is Noetherian.*

Proof. If R is a ring and I is an ideal in R , then any infinite ascending chain of ideals in the quotient R/I would correspond to an infinite chain of ideals in R , clearly a contradiction, since R is Noetherian. As for the second statement, by the First Isomorphism Theorem, $R/I \cong \text{im}(\rho)$, so it holds as well. \square

Theorem 3.1.3. *The following are equivalent:*

1. *R is a Noetherian ring.*
2. *Every nonempty set of ideals of R contains a maximal element under inclusion.*
3. *Every ideal of R is finitely generated.*

Proof. Assume R is Noetherian and let Σ be any nonempty collection of ideals in R . Choose $I_1 \in \Sigma$. If I_1 is a maximal ideal of R , then 2) holds, so assume I_1 is not a maximal ideal of R . Then there is $I_2 \in \Sigma$ such that $I_1 \subset I_2$. Then either I_2 is maximal, or there is an $I_3 \in \Sigma$ such that $I_2 \subset I_3$. Since any such chain is finite by the fact that R is a Noetherian ring, there must be some $k \in \mathbb{Z}^+$ such that $I_k \in \Sigma$ is maximal. Thus, 1) implies 2).

Assume that every nonempty set of ideals of a ring R contains a maximal element under inclusion. Let I be an ideal of R and let Σ be the collection of all finitely generated ideals of I . Since $\{0\} \in \Sigma$, this collection is nonempty. By assumption, this set contains a maximal element I' . If $I' \neq I$, let $x \in I - I'$. Since $I' \in \Sigma$, I' is finitely generated by assumption, the ideal generated by (I', x) is also finitely generated. This contradicts the maximality of I' , so $I = I'$ is finitely generated. So, 2) implies 3).

Assume that every ideal of R is finitely generated. Let $I_1 \subseteq \dots \subseteq I_3$ be a chain of ideals in R . Let $I = \bigcup_{i=1}^{\infty} I_i$ and note that I is an ideal. By 3), I is finitely generated, say by the elements a_1, \dots, a_n . Since $a_i \in I$ for all i , each a_i lies in one of the ideals in the chain, say I_{j_i} . Let $m = \max\{j_1, \dots, j_n\}$. Then $a_i \in I_m$ for all i so the ideal they generate is contained in I_m , which means $I \subseteq I_m$, which implies the two are equal, giving us $I_m = I = I_k$ for all $k \geq m$, which proves 1). Thus, 3) implies 1). \square

Corollary 3.1.4. *If R is a Principal Ideal Domain (hence, if R is a field or a Euclidean Domain) then R is a Noetherian ring.*

Proof. All Principal Ideal Domains satisfy 3) in Theorem 3.1.3. \square

3.2 Algebraic Geometry: Definitions and Basic Results

Definition 3.2.1. Two elements a, b in a ring R are said to **commute** if $a \times b = b \times a$. The **center** of a ring R is the set of all elements which *commute*

with every element of R ; the set $\{z \in R \mid zr = rz \text{ for all } r \in R\}$.

Let k be a field. A ring R is called a **k -algebra** if k is contained in the center of R and the identity of k is the identity of R .

Definition 3.2.2. The ring R is a **finitely generated algebra** if R is generated as a ring by k together with some set r_1, \dots, r_n of R . Let R and S be k -algebras. A map $\rho : R \rightarrow S$ is a **k -algebra homomorphism** if ρ is a ring homomorphism that is identity on k .

We've already dealt extensively with k -algebras; after all, a polynomial ring $R[x_1, \dots, x_n]$ is a R -algebra whenever R is a field.

Proposition 3.2.3. *The ring R is a finitely generated k -algebra if and only if there is some surjective k -algebra homomorphism $\rho : k[x_1, \dots, x_n] \rightarrow R$ from the polynomial ring in a finite number of variables onto R that is the identity map on k . Any finitely generated k -algebra is Noetherian.*

Proof. If R is generated as a k -algebra by r_1, \dots, r_n then we may define the map $\rho : k[x_1, \dots, x_n] \rightarrow R$ by $\rho(x_i) = r_i$ for all i and $\rho(a) = a$ for all $a \in k$. Then ρ extends uniquely to a surjective homomorphism.

Conversely, assume that there is some surjective k -algebra homomorphism $\rho : k[x_1, \dots, x_n] \rightarrow R$ from the polynomial ring in a finite number of variables onto R that is the identity map on k . Then the images of x_1, \dots, x_n under ρ generate R as a k -algebra, proving R is finitely generated.

As for the final statement, since $k[x_1, \dots, x_n]$ is Noetherian, any finitely generated k -algebra is therefore the quotient of a Noetherian ring, and hence is Noetherian by Corollary 3.1.2. \square

Definition 3.2.4. The set \mathbb{A}^n of n -tuples of elements of the field k is called the **affine n -space over k** . If x_1, \dots, x_n are viewed as independent variables over k , then the polynomials f in $k[x_1, \dots, x_n]$ can be seen as k -valued functions $f : \mathbb{A}^n \rightarrow k$ on \mathbb{A}^n by evaluating f at the points in \mathbb{A}^n such that $f : (a_1, \dots, a_n) \mapsto f(a_1, \dots, a_n) \in k$. This gives a ring of k -valued functions on \mathbb{A}^n , denoted $k[\mathbb{A}^n]$ and called the **coordinate ring of \mathbb{A}^n** .

For example, when $k = \mathbb{R}$ and $n = 2$, the coordinate ring of Euclidean 2-space \mathbb{R}^2 is denoted by $\mathbb{R}[\mathbb{A}^2]$ and is the ring of polynomials in two variables, say x and y , acting as real-valued functions on \mathbb{R}^2 .

At this point, recall the discussion in Section 1.5. Each subset S of functions in the coordinate ring $k[\mathbb{A}^n]$ determines a subset $\mathcal{Z}(S)$ of affine

space, the set of points where all functions in S are simultaneously zero. More formally, $\mathcal{Z}(S) = \{(a_1, \dots, a_n) \in \mathbb{A}^n : f(a_1, \dots, a_n) = 0 \text{ for all } f \in S\}$, where $\mathcal{Z}(\emptyset) = \mathbb{A}^n$. Subsets of affine space corresponding to subsets S have special names.

Definition 3.2.5. A subset V of \mathbb{A}^n is called an **affine algebraic set** or a **variety** if V is the set of common zeroes of some set S of polynomials; that is, if $V = \mathcal{Z}(S)$ for some $S \subseteq k[\mathbb{A}^n]$. Then $V = \mathcal{Z}(S)$ is called the **locus** of S in \mathbb{A}^n .

Definition 3.2.6. ([2], p 24) A **semi-algebraic subset of \mathbb{R}^n** is a subset of the form

$$\bigcup_{i=1}^s \bigcap_{j=1}^{r_i} \{x \in \mathbb{R}^n \mid f_{i,j} *_{i,j} 0\},$$

where $f_{i,j} \in \mathbb{R}[x_1, \dots, x_n]$ and $*_{i,j}$ is either $<$ or $=$, for $i = 1, \dots, s$ and $j = 1, \dots, r_i$. The semi-algebraic subsets of \mathbb{R}^n form the smallest family of subsets containing all sets of the form

$$\{x \in \mathbb{R}^n \mid f(x) > 0\}, \text{ where } f \in \mathbb{R}[x_1, \dots, x_n],$$

and are closed under taking finite intersections, finite unions, and complements. Also, if we call the conditions $f(x) > 0$, $f(x) < 0$, or $f(x) = 0$ *sign conditions on the polynomial f* , then a semi-algebraic subset of \mathbb{R}^n is defined by a boolean combination of sign conditions involving a finite number of polynomials.

Definition 3.2.7. A nonempty affine set V is called **irreducible** if it cannot be written as $V = V_1 \cup V_2$, where V_1 and V_2 are proper algebraic sets in V . An irreducible affine algebraic set is called an affine **irreducible variety**.

Though semi-algebraic sets and irreducible varieties will not be dealt with immediately, they are introduced here to emphasize the fact that we already have all the information we need to understand their basic definitions, if not their underlying theories.

Note that the locus of a single polynomial of the form $f - g$ is the same as the solutions in the affine n -space of the equation $f = g$.

Proposition 3.2.8. *Let S and T be subsets of $k[\mathbb{A}^n]$.*

1. *If $S \subseteq T$, then $\mathcal{Z}(T) \subseteq \mathcal{Z}(S)$. This means that \mathcal{Z} is inclusion reversing, or contravariant.*

2. $\mathcal{Z}(S) = \mathcal{Z}(I)$, where $I = (S)$ is the ideal in $k[\mathbb{A}^n]$ generated by S .
3. The intersection of two varieties is a variety: $\mathcal{Z}(S) \cap \mathcal{Z}(T) = \mathcal{Z}(S \cup T)$.
More generally, the arbitrary intersection of varieties is variety: if $\{S_j\}$ is any collection of subsets of $k[\mathbb{A}^n]$, then $\bigcap \mathcal{Z}(S_j) = \mathcal{Z}(\bigcup S_j)$.
4. The union of two varieties is a variety: $\mathcal{Z}(I) \cup \mathcal{Z}(J) = \mathcal{Z}(IJ)$, where I and J are two ideals and IJ is their product.
5. $\mathcal{Z}(0) = \mathbb{A}^n$ and $\mathcal{Z}(1) = \emptyset$, where 1 and 0 denote constant functions.

Since every ideal I in the Noetherian ring $k[x_1, \dots, x_n]$ is finitely generated, say $I = (f_1, \dots, f_q)$, it follows from Theorem 3.2.8 3) that $\mathcal{Z}(I) = \mathcal{Z}(f_1) \cap \mathcal{Z}(f_2) \cap \dots \cap \mathcal{Z}(f_q)$; each variety is the intersection of a finite number of hypersurfaces in \mathbb{A}^n . This geometric property in affine n -space is equivalent to the Hilbert's Basis Theorem in algebraic rings.

Given a set $V \subseteq \mathbb{A}^n$, there may be many such ideals I such that $V = \mathcal{Z}(I)$, but there is a unique largest ideal that determines V , and is given by the set of all polynomials that vanish on V . In general, define for any subset A of \mathbb{A}^n the mapping $\mathcal{I}(A) = \{f \in k[x_1, \dots, x_n] : f(a_1, \dots, a_n) = 0 \text{ for all } (a_1, \dots, a_n) \in A\}$. Then $\mathcal{I}(A)$ is an ideal, and the largest unique ideal of functions that are identically zero on A , defining a correspondence $\mathcal{I} : \{\text{subsets in } \mathbb{A}^n\} \rightarrow \{\text{ideals of } k[\mathbb{A}^n]\}$.

Proposition 3.2.9. *Like \mathcal{Z} , \mathcal{I} also has some basic properties, and relations exist between the two correspondences.*

1. If $A \subseteq B$ then $\mathcal{I}(B) \subseteq \mathcal{I}(A)$, so \mathcal{I} is also contravariant.
2. $\mathcal{I}(A \cup B) = \mathcal{I}(A) \cap \mathcal{I}(B)$.
3. $\mathcal{I}(\emptyset) = k[x_1, \dots, x_n]$, and if k is infinite, then $(\mathbb{A}^n) = 0$.
4. If A is any subset of \mathbb{A}^n , then $A \subseteq \mathcal{Z}(\mathcal{I}(A))$, and if I is any ideal, then $I \subseteq \mathcal{I}(\mathcal{Z}(I))$.
5. If $V = \mathcal{Z}(I)$ is a variety then $V = \mathcal{Z}(\mathcal{I}(V))$ and if $I = \mathcal{I}(A)$ then $\mathcal{I}(\mathcal{Z}(I)) = I$. In other words, $\mathcal{Z}(\mathcal{I}(\mathcal{Z}(I))) = \mathcal{Z}(I)$ and $\mathcal{I}(\mathcal{Z}(\mathcal{I}(A))) = \mathcal{I}(A)$.

Thus, \mathcal{I} and \mathcal{Z} are inverses, provided we restrict ourselves to collection of varieties $V = \mathcal{Z}(I)$ in \mathbb{A}^n and the sets of ideals in $k[\mathbb{A}^n]$ of the form $\mathcal{I}(V)$. When k is algebraically closed, we characterize those ideals I that are of the form $\mathcal{I}(V)$ for some variety V in terms of just ring properties of I .

Definition 3.2.10. If $V \subseteq \mathbb{A}^n$ is a variety, the quotient ring $k[\mathbb{A}^n]/\mathcal{I}(V)$ is called the **coordinate ring** of V and is denoted $k[V]$. Note that for $V = \mathbb{A}^n$ and k is infinite, then $\mathcal{I}(V) = 0$, in which case the previous terminology of $k[\mathbb{A}^n]$ still holds.

An interesting consequence of the coordinate ring is that we can have two polynomial functions that are distinct in $k[\mathbb{A}^n]$ but the same in $k[V]$. Polynomials in $k[\mathbb{A}^n]$ define k -valued functions on V by restricting these functions on \mathbb{A}^n to the subset V . Then two polynomial functions f and g define the same function on V if and only if $f - g \in \mathcal{I}(V)$, and $f - g = 0$ on V . The elements of $k[V]$, which are just cosets $\bar{f} = f + \mathcal{I}(V)$, are the restrictions to V of ordinary polynomial functions from \mathbb{A}^n to k .

Example 3.2.11. Let $V = \mathcal{Z}(xy - 1)$ be the hyperbola $y = 1/x$ in \mathbb{R}^2 , so that $\mathbb{R}[V] = \mathbb{R}(x, y)/(xy - 1)$. Then polynomial $f(x, y) = x$ (the x -coordinate function) and the polynomial $g(x, y) = x + (xy - 1)$ define the same function in V , even though they define different functions on \mathbb{R}^2 . For any $(x_1, y_1) \in V$, $f(x_1, y_1) = g(x_1, y_1)$. In the quotient ring $\mathbb{R}[V]$ we have $\overline{xy} = 1$, so $\mathbb{R}[V] \cong \mathbb{R}[x, 1/x]$ for any function $\bar{f} \in \mathbb{R}[V]$ and any $(a, b) \in V$ we have $\bar{f}(a, b) = f(a, 1/a)$ for any polynomial $f \in k[x, y]$ that maps to \bar{f} in the quotient.

Suppose $V \subseteq \mathbb{A}^n$ and $W \subseteq \mathbb{A}^m$ are two varieties. Since V and W are defined by the zeroes of polynomials, it follows that mappings between the two are also defined by polynomials, like so:

Definition 3.2.12. A map $\rho : V \rightarrow W$ is called a **morphism** (or a **polynomial map** or **regular map**) of algebraic sets if there are polynomials $\rho_1, \dots, \rho_m \in k+$ such that $\rho((a_1, \dots, a_n)) = (\rho_1(a_1, \dots, a_n), \dots, \rho_m(a_1, \dots, a_n))$ for all $(a_1, \dots, a_n) \in V$. The map $\rho : V \rightarrow W$ is an **isomorphism** of algebraic sets if there exists an inverse morphism, say φ , such that its composition with ρ gives either the identity mapping on V or the identity mapping on W , depending on which way it is composed; $\varphi : W \rightarrow V, \rho \circ \varphi = 1_W$ and $\varphi \circ \rho = 1_V$.

For an example of such a mapping, see ([10], p15) (The mapping π is technically a mapping of semi-algebraic sets, not varieties, but the principle is the same). In this case, the fundamental invariants serve as the ρ_1, \dots, ρ_m .

The choice of ϕ_1, \dots, ϕ_m is not necessarily unique. Returning to Example 3.2.11, $f = x$ and $g = x + (xy - 1)$ define the same morphism from $V = \mathcal{Z}(xy - 1)$ to $W = \mathbb{A}^1$.

Proposition 3.2.13. *Let $V \subseteq \mathbb{A}^n$ and $W \subseteq \mathbb{A}^m$ be varieties. Then there is a bijective correspondence*

$$\{\text{morphisms from } V \text{ to } W\} \leftrightarrow \{k\text{-algebra homomorphisms from } k[W] \text{ to } k[V]\}.$$

More elaborately,

1. *Every morphism $\rho : V \rightarrow W$ induces an associated k -algebra homomorphism $\tilde{\rho} : k[W] \rightarrow k[V]$ defined by $\tilde{\rho}(f) = f \circ \rho$.*
2. *Every k -algebra homomorphism $\Phi : k[W] \rightarrow k[V]$ is induced by a unique morphism $\rho : V \rightarrow W$ ($\Phi = \tilde{\rho}$).*
3. *If $\rho : V \rightarrow W$ and $\varphi : W \rightarrow U$ are morphisms of varieties, then $\varphi \circ \rho = \tilde{\varphi} \circ \tilde{\rho} : k[U] \rightarrow k[V]$.*
4. *the morphism $\rho : V \rightarrow W$ is an isomorphism if and only if $\tilde{\rho} : k[W] \rightarrow k[V]$ is a k -algebra homomorphism.*

Proof. 1. Suppose $F \in k[x_1, \dots, x_m]$. Then $F \circ \rho = f(\rho_1, \dots, \rho_m) \in k[x_1, \dots, x_n]$ since ρ_1, \dots, ρ_m are polynomials in x_1, \dots, x_n . Now suppose $F \in \mathcal{I}(W)$. Then $F \circ \rho((a_1, \dots, a_n)) = 0$ for every (a_1, \dots, a_n) , since $\rho((a_1, \dots, a_n)) \in W$. This means $F \circ \rho \in \mathcal{I}(V)$. It then follows that ρ induces a well-defined map from the quotient ring $k[x_1, \dots, x_m]/\mathcal{I}(W)$ to the quotient ring $k[x_1, \dots, x_n]/\mathcal{I}(V)$:

$$\begin{aligned} \tilde{\rho} : k[W] &\rightarrow k[V], \\ f &\mapsto f \circ \rho \end{aligned}$$

where $f \circ \rho$ is given by $F \circ \rho = \mathcal{I}(V)$ for any polynomial $F = F(x_1, \dots, x_m)$ with $f = F + \mathcal{I}(W)$. Then $\tilde{\rho}$ is a k -algebra homomorphism. Also, the morphism from V to W has induced a k -algebraic homomorphism from $k[W]$ to $k[V]$. This proves 1).

2. Conversely, suppose that Φ is any k -algebra homomorphism from the coordinate ring $k[W] = k[x_1, \dots, x_m]$ to $k[V] = k[x_1, \dots, x_n]/\mathcal{I}(V)$. Let F_i be representative in $k[x_1, \dots, x_m]$ for the image under Φ of $\bar{x}_i \in k[W]$. Then $\rho(F_1, \dots, F_m)$ defines a polynomial mapping from \mathbb{A}^n to \mathbb{A}^m and in fact ρ is a morphism from V to W . To see this, it suffices to check that ρ maps a point of V to a point of W , since ρ is already defined

by polynomials. If $g \in \mathcal{I}(W) \subset k[x_1, \dots, x_m]$, then in $k[W]$ we have $g(x_1 + \mathcal{I}(W), \dots, x_m + \mathcal{I}(W)) = g(x_1, \dots, x_m) + \mathcal{I}(W) = \mathcal{I}(W) = 0 \in k[W]$, so $\Phi(g(x_1 + \mathcal{I}(W), \dots, x_m + \mathcal{I}(W))) = 0 \in k[V]$. And since Φ is a k -algebra homomorphism, it further follows that

$$g(\Phi(x_1 + \mathcal{I}(W)), \dots, \Phi(x_m + \mathcal{I}(W))) = 0 \in k[V]$$

By its definition,

$$\Phi(x_i + \mathcal{I}(W)) = F_i \text{ mod } \mathcal{I}(V)$$

so

$$g(F_1 \text{ mod } \mathcal{I}(V), \dots, F_m \text{ mod } \mathcal{I}(V)) = 0 \in k[V]$$

(where $F_i \text{ mod } \mathcal{I}(V)$ just means we find the coset of F_i in the quotient $k[V]$). As a result, we can say that

$$g(F_1, \dots, F_m) \in \mathcal{I}(V)$$

What all of this shows is that if $(a_1, \dots, a_n) \in V$ then every polynomial in $\mathcal{I}(W)$ vanishes on $\rho(a_1, \dots, a_n)$. By 5) in Theorem 3.2.9, this means that $\rho(a_1, \dots, a_n) \in \mathcal{Z}(\mathcal{I}(W)) = W$, which proves that ρ maps a point in V to a point in W . Since all the cosets of F_i are well-defined, this morphism does not depend on the choice of F_i . Furthermore, ρ induces the original k -algebra homomorphism Φ from $k[W]$ to $k[V]$, so $\tilde{\rho} = \Phi$, as both take $F_i + \mathcal{I}(V)$ on $x_i = \mathcal{I}(W) \in k[W]$.

3. This proof applies Statement 1) and 2). 1) defines what a $\tilde{\rho}$ mapping does, and 2) states that for each $\tilde{\rho}$, the ρ that induced it is unique. So all we have to do is show that $\widetilde{\varphi \circ \phi}$ and $\widetilde{\phi} \circ \tilde{\varphi}$ are induced by the same morphism $\varphi \circ \rho$. Let $f \in k[U]$. Then

$$\widetilde{\phi} \circ \tilde{\varphi}(f) = \tilde{\rho} \circ (f \circ \varphi) = (f \circ \varphi) \circ \rho = f \circ (\varphi \circ \rho) = \widetilde{\varphi \circ \rho}(f),$$

so the two are equivalent, and have the same unique associated morphism.

4. \rightarrow . Suppose $\rho : V \rightarrow W$ is an isomorphism, with inverse ρ^{-1} . Then, by 1), both ρ and ρ^{-1} induce associated k -algebra homomorphisms, $\tilde{\rho}$ and $\tilde{\rho}^{-1}$. Then, by 3),

$$\widetilde{\rho \circ \rho^{-1}} = \tilde{1}_W = 1_{k[W]} = \tilde{\rho} \circ \widetilde{\rho^{-1}},$$

(where 1 denotes some identity mapping with the domain specified by the subscript) and the vice-versa holds as well, so $\tilde{\rho}$ is an isomorphism, proving the first part.

\leftarrow . Conversely, assume that $\tilde{\rho}$, which maps $k[W]$ to $k[V]$ is an isomorphism with the inverse $\tilde{\rho}^{-1}$. Let $\tilde{\rho}^{-1}$ have the corresponding unique morphism $\tilde{\varphi}$. Then, by 3),

$$\widetilde{\varphi \circ \rho} = \tilde{\rho} \circ \tilde{\varphi} = \tilde{\rho} \circ \tilde{\rho}^{-1} = 1_{k[V]}.$$

Since the identity mapping on $k[V]$ is induced by the unique morphism 1_V , this implies that $\varphi \circ \rho = 1_V$, and, by similar logic, $\rho \circ \varphi = 1_W$, which means that ρ is an isomorphism, completing the proof. \square

For an example of such a morphism and its induced homomorphism, refer again to [10] (And again, remember that the set in this paper is semi-algebraic, not a variety.).

Thanks to Proposition 3.2.13, we can now define a morphism in terms of V and W without making reference to the affine spaces \mathbb{A}^n and A^m containing them.

Corollary 3.2.14. *Suppose $\rho : V \rightarrow W$ is a map of varieties. Then ρ is a morphism if and only if for every $f \in k[W]$ the composite map $f \circ \rho$ is an element of $k[V]$ as a k -valued function on V . When ρ is a morphism, $\rho(v) = w$ with $v \in V$ and $w \in W$ if and only if $\tilde{\rho}^{-1}(\mathcal{I}(\{v\})) = \mathcal{I}(\{w\})$.*

Proof. First, we'll show that if ρ is any map from V to W such that $\tilde{\rho}$ is a k -algebra homomorphism then $\rho(v) = w$ if and only if $\tilde{\rho}^{-1}(\mathcal{I}(\{v\})) = \mathcal{I}(\{w\})$, proving the second statement. $\rho(v) = w \leftrightarrow$ every polynomial f vanishing at w vanishes at $\rho(v)$ (which follows from Proposition 3.2.9, $\{w\} = \mathcal{Z}(\mathcal{I}(\{w\})) \leftrightarrow \tilde{\rho}(f)$ vanishes at $v \leftrightarrow \tilde{\rho}(f) \in \mathcal{I}(\{v\})$ for every $\mathcal{I}(\{w\}) \leftrightarrow \tilde{\rho}(\mathcal{I}(\{w\})) \subseteq \mathcal{I}(\{v\})$ or $\mathcal{I}(\{w\}) \subseteq \tilde{\rho}^{-1}(\mathcal{I}(\{v\}))$). Since $\mathcal{I}(\{w\})$ and $\mathcal{I}(\{v\})$ are both maximal ideals, the containment must be strict: $\tilde{\rho}^{-1}(\mathcal{I}(\{v\})) = \mathcal{I}(\{w\})$.

We now prove the first statement. If ρ is a morphism, then $f \circ \rho \in k[V]$ for every $f \in k[W]$, just by the definition of what a morphism does. For the converse, observe that composition with any map $\rho : V \rightarrow W$ defines a k -algebra homomorphism from $k[W]$ to $k[V]$. If $f \circ \rho \in k[V]$ for every $f \in k[W]$, then $\tilde{\rho}$ is a k -algebra homomorphism from $k[W]$ to $k[V]$, so by Proposition 3.2.9, $\tilde{\rho} = \tilde{\varphi}$ for a unique morphism $\varphi : V \rightarrow W$. Also, since $\tilde{\rho}$ is a k -algebra

homomorphism from $k[W]$ to $k[V]$ it follows that $\rho(v) = w \leftrightarrow \tilde{\rho}^{-1}(\mathcal{I}(\{v\})) = \mathcal{I}(\{w\})$, by the second statement already proven. Because $\tilde{\rho} = \tilde{\varphi}$, this is equivalent to saying that $\varphi^{-1}(\mathcal{I}(\{v\})) = \mathcal{I}(\{w\})$, and so $\varphi(v) = w$. Hence ρ and φ define the same map on V and hence ρ is a morphism. \square

3.3 Computations in Affine Algebraic Sets and k -Algebras

Just as we've studied methods of obtaining computational results in polynomial rings, we will now turn our attention to finding computational results for k -algebras. From our previous results with Gröbner bases (see Chapter 1, if a refresher is required), we know that since $\mathcal{I}(V)$ is an ideal in $k[\mathbb{A}^n]$, it is finitely generated and hence has a Gröbner basis, which can be used to calculate a unique remainder for any polynomial $f \in k[\mathbb{A}^n]$. This remainder also serves a unique representative for the coset \bar{f} in the coordinate ring $k[\mathbb{A}^n]/\mathcal{I}(V)$. In Section 1.5, we looked at how Gröbner bases and elimination theory could be applied to solving system of algebraic equations. The same theory can be used to determine explicitly a set of generators for the kernel of a k -algebra homomorphism

$$\Phi : k[y_1, \dots, y_m]/J \rightarrow k[x_1, \dots, x_n]/I$$

where I and J are ideals. (Later, we'll look at the particular case when $I = \mathcal{I}(V)$ and $J = \mathcal{I}(W)$, are the ideals associated to affine algebraic sets $V \subseteq \mathbb{A}^n$ and $W \subseteq \mathbb{A}^m$, so that Φ corresponds to a morphism from V to W .)

(For the proceeding, let \bar{y} denote the mapping of some polynomial y from an algebra into a quotient ring, where the algebra and quotient ring are clear from the context.)

Algorithm 3.3.1. (Finding the Kernel of a k -algebra Homomorphism)

1. Input: generators h_1, \dots, h_t of an ideal $J \subseteq k[y_1, \dots, y_m]$, generators f_1, \dots, f_s of an ideal $I \subseteq k[x_1, \dots, x_n]$, a k -algebra homomorphism Φ such that

$$\Phi : k[y_1, \dots, y_m]/J \rightarrow k[x_1, \dots, x_n]/I.$$

2. Set $i := 1$. Send y_i to Algorithm 1.3.9 as f , and h_1, \dots, h_t as b_1, \dots, b_m . Set $\bar{y}_i = r$. Set $\Phi(\bar{y}_i) = \phi_i$. Let $i := i+1$. Repeat while $i \leq m$.

3. Calculate the reduced Gröbner basis B of the ideal $\mathcal{A} = (y_1 - \phi_1, \dots, y_m - \phi_m, f_1, \dots, f_s)$, with respect to the lexicographic monomial ordering $x_1 > \dots > x_n > y_1 > \dots > y_m$.
4. Let $B' = B \cap k[y_1, \dots, y_m]$.
5. Let $k := |B'|$, $B' = \{b_1', \dots, b_k'\}$. Set $i := 1$. Send b_i' to Algorithm 1.3.9, with $f = b_i'$ and h_1, \dots, h_t as the b_1, \dots, b_m . Replace b_i' with r . Let $i := i + 1$. Repeat while $i \leq k$. Then END.
6. Output: the generators b_i' of $\ker\Phi$.

Proof. If we can show that $\ker\phi = (\mathcal{A} \cap k[y_1, \dots, y_m])/J$, then the rest of the algorithm follows by Proposition 1.5.5. Suppose that $f \in \mathcal{A} \cap k[y_1, \dots, y_m]$. That means that $f \in k[y_1, \dots, y_m]$ and that $f \in \mathcal{A}$. The latter containment for f implies that it can also be decomposed as follows. If f_1, \dots, f_s are generators for I in $k[x_1, \dots, x_n]$, then

$$f(y_1, \dots, y_m) = \sum_{i=1}^n a_i(y_i - \phi_i) + \sum_{j=1}^s b_j f_j$$

as polynomials in $R = [y_1, \dots, y_m, x_1, \dots, x_n]$, where $a_1, \dots, a_n, b_1, \dots, b_s \in R$. Substituting $y_i = \phi_i$ we see that $f(\phi_1, \dots, \phi_m) \in I$. Since $\Phi(\bar{f}) = (f(\phi_1, \dots, \phi_m))/I$, it follows that f represents a coset in the kernel of Φ . Conversely, suppose $f \in k[y_1, \dots, y_m]$ represents an element in $\ker\Phi$. Then $f(\phi_1, \dots, \phi_m) \in I$ and so $f(\phi_1, \dots, \phi_m) \in \mathcal{A}$. Since $y_i - \phi_i \in \mathcal{A}$,

$$f(y_1, \dots, y_m) \equiv f(\phi_1, \dots, \phi_m) \equiv \bar{0},$$

, when viewed as a coset in the quotient of \mathcal{A} , so $f \in \mathcal{A} \cap k[y_1, \dots, y_m]$. \square

We can also tell a few things about the elements in the image of ϕ :

Proposition 3.3.2. *Under the previous notation, if $f \in k[x_1, \dots, x_n]$, then \bar{f} is in the image of ϕ if and only if the remainder after general polynomial division of f by the elements in B is an element $h \in k[y_1, \dots, y_m]$ in which case $\Phi(\bar{h}) = \bar{f}$.*

Proof. \rightarrow . Suppose that $f \in k[x_1, \dots, x_n]$ represents an element in the image of Φ , that $\bar{f} = \Phi\bar{h}$ for some $h \in k[y_1, \dots, y_m]$. Then

$$f(x_1, \dots, x_n) - h(\phi_1, \dots, \phi_m) \in I$$

as polynomials in $k[x_1, \dots, x_n]$ and so $f(x_1, \dots, x_n) - h(\phi_1, \dots, \phi_m) \in \mathcal{A}$ as polynomials in R . As before, since each $y_i - \rho_i \in \mathcal{A}$, it follows that

$$f(x_1, \dots, x_n) - h(y_1, \dots, y_m) \in \mathcal{A},$$

Then $f(x_1, \dots, x_n)$ and $h(y_1, \dots, y_m)$ leave the same remainder after general polynomial division by the elements in B . Since $x_1 > \dots > x_n > y_1 > \dots > y_m$, the remainder of $h(y_1, \dots, y_m)$ is again a polynomial h_0 only involving y_1, \dots, y_m . Note also that $h - h_0 \in \mathcal{A} \cap k[y_1, \dots, y_m]$ so \bar{h} and $\overline{h_0}$ differ by an element in $\ker \Phi$, so $\Phi(\overline{h_0}) = \Phi(\bar{h}) = \bar{f}$.

←. If f leaves the remainder $h \in k[y_1, \dots, y_m]$ after general polynomial division by the elements in B then $f(x_1, \dots, x_n) - h(y_1, \dots, y_m) \in \mathcal{A}$:

$$f(x_1, \dots, x_n) - h(y_1, \dots, y_m) = \sum_{i=1}^n a_i(y_i - \rho_i) + \sum_{j=1}^s b_j f_j$$

as polynomials in R , where $a_1, \dots, a_n, b_1, \dots, b_s \in R$. Substituting $y_i = \phi_i$ we obtain

$$f(x_1, \dots, x_n) - h(\rho(1), \dots, \rho_m) \in I$$

as polynomials in x_1, \dots, x_n and so $\bar{f} = \Phi(\bar{h})$. □

A corollary follows.

Corollary 3.3.3. *The map Φ above is surjective if and only if for each $i, 1 \leq i \leq n$, the reduced Gröbner basis B contains a polynomial $x_i - h_i$ where $h_i \in k[y_1, \dots, y_m]$, under the notation used previously. (Note: this shouldn't be too hard to program)*

Proof. By the algorithm, Φ is a surjective homomorphism if and only if for each $i = 1, 2, \dots, n$, dividing x_i by the elements in the Gröbner basis B leaves a remainder $h_i \in k[y_1, \dots, y_m]$. In particular, $x_n - h_n$ leaves a remainder of 0. But then the leading term of some element $b_n \in B$ divides the leading term of $x_n - h_n$, and since $x_1 > \dots > z_n > y_1 \dots y_m$ by the choice of the ordering, the leading term of $x_n - h_n$ is x_n . Then $LT(b_n) = x_n$ and so $b_n = x_n - h_{n,0} \in B$ for some $h_{n,0} \in k[y_1, \dots, y_m]$. (And, in fact, $h_{n,0}$ is the remainder of h_n , after division by the elements of B) An inductive argument completes the proof. Since $x_{n-1} - h_{n-1}$ leaves a remainder of 0, there is an element b_{n-1} whose leading term is x_{n-1} . since B is a reduced Göbner basis and $b_n \in B$, the leading term of b_n (x_n), does not divide any of the terms in b_{n-1} , it follows that $b_{n-1} = x_{n-1} - h_{n-1,0} \in G$ for some $h_{n-1,0} \in k[y_1, \dots, y_m]$. We continue in a similar vein until we reach x_1 , completing the proof. □

3.4 Gröbner Bases and their Varieties

When a variety is a finite set, we can actually use Gröbner bases to place a bound on the number of elements in the variety. The process is fairly simple, but the proof is very complicated, and will require some preliminary results.

Theorem 3.4.1. (Theorem 4.56 (c), [1]) *Let $I \subseteq \mathbb{C}[x_1, \dots, x_n]$ be an ideal. The number of elements in the variety $\mathcal{Z}(I)$ is less than or equal to the dimension of $\mathbb{C}[x_1, \dots, x_n]/I$ as a \mathbb{C} -vector space.*

This deceptively simple theorem will require a few more concepts to prove.

Definition 3.4.2. Recall Definition 1.1.5. A field F is said to be **algebraically closed** if any non-constant single variable polynomial $p(x) \in F[x]$ with coefficients in F has a solution in F .

Note that the fact that \mathbb{C} is an algebraically closed field is absolutely essential to the proof; in fact, the theorem holds for any field F that is contained within an algebraically closed field. Something else we will need (without proving) is Hilbert's Nullstellensatz.

Theorem 3.4.3. Hilbert's Nullstellensatz (in the field \mathbb{C}) ([1], p 119) *Let I be an ideal of $\mathbb{C}[x_1, \dots, x_n]$. If a polynomial p with coefficients in \mathbb{C} vanishes on $\mathcal{Z}(I)$, then $p^N \in I$ for some N .*

Statement 3.4.4. $\mathbb{C}[x_1, \dots, x_n]/I$ is of finite dimension if and only if $\mathcal{Z}(I)$ is a non-empty finite set.

Proof. \rightarrow If $\mathbb{C}[x_1, \dots, x_n]/I$ is a finite dimensional vector space of dimension N over \mathbb{C} , then the set $\{1, x_1, \dots, x_1^N\}$ is linearly dependent in $\mathbb{C}[x_1, \dots, x_n]/I$. As a result, there is a polynomial $p_1(x_1)$ of degree at most N in the ideal I . The first coordinate for any $x \in \mathcal{Z}(I)$ must be a root of p_1 . Doing the same for all the variables, we see that $\mathcal{Z}(I)$ is a finite set, as there are only finitely many possibilities for solutions to the polynomials in I .

\leftarrow Assume $\mathcal{Z}(I)$ is a finite set. Take a polynomial $p_1(x_1) \in \mathbb{C}[x_1]$ whose roots are the first coordinates of the elements of $\mathcal{Z}(I)$. By Theorem 3.4.3, $p_1^N \in I$ for some $N \in \mathbb{N}$. Doing the same for all the variables, we see that for every i , there exists a polynomial of degree d_i in $\mathbb{C}[x_i]$ that is also in the ideal I . It follows that $\mathbb{C}[x_1, \dots, x_n]/I$ has a basis consisting of monomials whose degrees in x_i is less than d_i . Thus, $\mathbb{C}[x_1, \dots, x_n]/I$ is finite dimensional over K . \square

We now have the result that a finite dimension for $\mathbb{C}[x_1, \dots, x_n]/I$ implies a finite variety, and vice versa. The first part of the proof for Statement 3.4.4 contains the start of the proof for Theorem 3.4.1. As mentioned, for each i where $1 \leq i \leq n$, there exists a polynomial $p_i(x_i)$ of degree at most N , which means each p_i has at most N roots, and hence there are at most N possibilities for each i -th coordinate in $\mathcal{Z}(I)$. This creates a lower and upper bound for the number of solutions in $\mathcal{Z}(I)$; $1 \leq |\mathcal{Z}(I)| \leq n * N$. But frankly, we can do better with that upper bound.

Definition 3.4.5. An element $m \in \mathbb{C}[x_1, \dots, x_n]/I$ is **separating** for I if m has distinct values at distinct elements of $\mathcal{Z}(I)$.

Lemma 3.4.6. *Separating elements always exist when $\mathbb{C}[x_1, \dots, x_n]/I$ is finite dimensional; that is, if $|\mathcal{Z}(I)| = N$, then at least one of*

$$m_i = x_1 + ix_2 + \dots + i^{n-1}x_n, m_i \in \mathbb{C}[x_1, \dots, x_n]/I$$

for $0 \leq i \leq (k-1) \binom{n}{2}$ is separating.

Proof. Let $x = (x_1, \dots, x_n), y = (y_1, \dots, y_n)$ be two distinct points of $\mathcal{Z}(I)$ and let $\mathcal{L}(x, y)$ be the number of $i, 0 \leq i \leq (k-1) \binom{n}{2}$, such that $m_i(x) = m_i(y)$. Since the polynomial

$$(x_1 - y_1) + (x_2 - y_2)T + \dots + (x_k - y_k)T^{k-1}$$

is not identically zero, it has no more than $k-1$ distinct roots. Hence $\mathcal{L}(x, y) \leq k-1$. As the number of 2-element subsets of $\mathcal{Z}(I)$ is $\binom{n}{2}$, the lemma follows. \square

We can use these separating polynomials to our advantage.

Lemma 3.4.7. *If a is separating and $\mathcal{Z}(I)$ has N elements, then $\{1, m, \dots, m^{n-1}\}$ is linearly independent in $\mathbb{C}[x_1, \dots, x_n]/I$.*

Proof. Prove by contradiction. Suppose there exists $c_i \in \mathbb{C}$ such that

$$\sum_{i=0}^{n-1} c_i m^i = 0$$

in $\mathbb{C}[x_1, \dots, x_n]$, which means the polynomial $\sum_{i=0}^{n-1} c_i m^i \in I$. thus, for all $x \in \mathcal{Z}(I)$,

$$\sum_{i=0}^{n-1} c_i m^i(x) = 0.$$

But then the single variable polynomial $\sum_{i=0}^{n-1} c_i T^i = 0$ has n distinct roots, and is therefore identically zero, which contradicts the fact that a is separating, by Lemma 3.4.6. Therefore, $\{1, m, \dots, m^{n-1}\}$ is a linearly independent set in $\mathbb{C}[x_1, \dots, x_n]/I$. \square

Theorem 3.4.1 follows clearly from this result.

Proof. (Theorem 3.4.1) By Statement 3.4.4, if $\mathbb{C}[x_1, \dots, x_n]/I$, then $\mathcal{Z}(I)$ is a finite set. By Lemma 3.4.7 and Lemma 3.4.6, if $\mathcal{Z}(I) = N$, and m is a separating element of $\mathbb{C}[x_1, \dots, x_n]/I$, then $\{1, m, \dots, m^{N-1}\}$ is a linearly independent set of N elements. That means any basis of $\mathbb{C}[x_1, \dots, x_n]/I$ has to have at least N elements, or possibly more, so we have $|\mathcal{Z}(I)| \leq \dim(\mathbb{C}[x_1, \dots, x_n]/I)$. \square

In order to actually calculate the bound on the number of elements of $\mathcal{Z}(I)$, we'll have to find a way to explicitly calculate a vector-space basis for $\mathbb{C}[x_1, \dots, x_n]/I$. Luckily, that's the easy part.

Definition 3.4.8. Recall the definition of the ideal of leading terms, 1.4.1. Suppose you have a given ideal I and the ideal of its leading terms, $LT(I)$. The monomials $m \notin LT(I)$ are called **standard** and the monomials $m \in LT(I)$ are called **non-standard**.

Theorem 3.4.9. (*[13], p 11*) Let I be any ideal and " $<$ " any monomial order on $\mathbb{C}[x_1, \dots, x_n]$. The set of (residue classes of) standard monomials is a \mathbb{C} -vector space basis for the quotient ring $\mathbb{C}[x_1, \dots, x_n]/I$.

Proof. Let B be a Gröbner basis for I , and consider the following algorithm which computes the normal form with respect to I .

Algorithm 3.4.10. Normal Form

1. Input: $p \in \mathbb{C}[x_1, \dots, x_n]$.
2. Check whether all monomials in p are standard. If so, END; $\bar{p} = p$.
(The polynomial is unchanged in the quotient ring $\mathbb{C}[x_1, \dots, x_n]/I$.)

3. Otherwise let $hnst(p)$ be the highest non-standard monomial occurring in p . Find $b \in B$ such that $LT(b)$ divides $hnst(p)$, say $m \cdot LT(b) = hnst(p)$.
4. replace p by $\bar{p} := p - m \cdot g$ and go to Step 2.

By Step 4, we have $LT(\bar{p}) < LT(p)$, and hence the fact that there is no infinite descending chain of monomials for a monomial ordering on $\mathbb{C}[x_1, \dots, x_n]$ allows us to say that the algorithm terminates with a representation of $p \in \mathbb{C}[x_1, \dots, x_n]$ as a \mathbb{C} -linear combination of standard monomials in the quotient ring $\mathbb{C}[x_1, \dots, x_n]/I$. Every such representation is necessarily unique, because, as we've already seen, the remainder r is unique. Thus, zero cannot be written as a non-trivial linear combination of standard monomials in $\mathbb{C}[x_1, \dots, x_n]/I$, and so the standard monomials form a linearly independent set in $\mathbb{C}[x_1, \dots, x_n]/I$. \square

Note that this result was hinted at by the second part of the proof for Statement 3.4.4.

The method for calculating the bound boils down to this: given the generators of an ideal $I \subset \mathbb{C}[x_1, \dots, x_n]$, we calculate its reduced Gröbner basis. Using the Gröbner basis, we count up all the standard monomials; the number of standard monomials is the upper bound of the number of points in $\mathcal{Z}(I)$.

Corollary 3.4.11. *If the reduced Gröbner basis for a finitely generated ideal $I \subset \mathbb{C}[x_1, \dots, x_n]/I$ is a set of polynomials such that the set of leading terms of all the monomials does not contain a term of the form x_j^i for all $1 \leq j \leq n$ and $i \in \mathbb{N}$, then $\mathcal{Z}(I)$ has infinitely many elements. Also, if the monomial ordering used to form the Gröbner basis was the lexicographic monomial ordering $x_1 > x_2 > \dots > x_n$, then elimination ideal I_j is nonempty for every j .*

Proof. If the set of leading terms of all the monomials does not contain a term of the form x_j^i for all $1 \leq j \leq n$ and $i \in \mathbb{N}$, then every power of x_j is in the set of standard monomials, creating an infinite set of standard monomials, which in turn means there are infinitely many solutions to $\mathcal{Z}(I)$. As for the second part, if there is a polynomial with the leading term x_j^i in the Gröbner basis of I , then it is certainly contained in I_j . \square

Note that the following statement also holds: if the set of leading terms does contain terms of the form above, then the variety is finite. In this case, we have a bound for each element in the multidegree (a_1, \dots, a_n) of possible standard monomials $x_1^{a_1} \dots x_n^{a_n}$, which in turn limits the number of possible standard monomials. Combining this fact with the second statement of the lemma, this means we can always use the method described in Section 1.5 to explicitly find the points of $\mathcal{Z}(I)$.

We will look at some examples using Theorem 3.4.1 in the next section; while it is a fairly minor theorem in the grand scheme of things, it provides a quick rule-of-thumb approach to analysing a variety.

3.5 Algebraic Computation and Invariance

(This section has been adapted from Section 2.6 of [13].) Given the problem of finding the zeroes of an ideal $I \subset \mathbb{C}[x_1, \dots, x_n]$, the standard solution is to compute a lexicographic Gröbner basis for I , as we did in Section 1.5. But this computation can be very time-consuming. In fact, the computation is especially time-consuming if the given set of generators for I are invariant under some finite group action, which, due to symmetry, happens quite often. Our goal here is to observe some applications of invariance that help, rather than hinder, computations in algebraic geometry.

First, we need to fix the notion of invariance in algebraic geometry, and establish some terminology.

Definition 3.5.1. Recall Definition 2.1.2. Using that notation, the **orbit** of a point $a \in A$ is the set $\{g \cdot a \mid g \in G\}$, and is sometimes called the **G-orbit of a**.

Note that the group G can act on an affine space \mathbb{A}^n just as easily as it can act on a polynomial ring $k[\mathbb{A}^n]$. This notion leads to our next definition.

Definition 3.5.2. The set of all G -orbits in \mathbb{C}^n is denoted \mathbb{C}^n/G and called the **orbit space** of G . We then have an induced action on the *coordinate ring* of \mathbb{C}^n , which is the polynomial ring $\mathbb{C}[x_1, \dots, x_n]$. The invariant subring $\mathbb{C}[x_1, \dots, x_n]^G$ consists of all polynomials which are fixed under the action of G . We know that this subring is finitely generated by h_1, \dots, h_m fundamental invariants. In geometric terms, the choice of these invariants amounts to choosing an embedding of the orbit space \mathbb{C}^n/G as an algebraic subvariety

into affine m -space \mathbb{C}^m . In that case, $\mathbb{C}^n/G \subseteq \mathbb{C}^m$ is referred to as an **orbit variety** and the equations that define it are called **syzygies** or algebraic relations among the h_i .

Using Gröbner bases to compute the syzygies of fundamental invariants is called *preprocessing*. An algorithm for the preprocessing follows.

Algorithm 3.5.3. (Preprocessing a fixed group G)

1. Input: Any finite matrix group G .
2. Compute a fundamental set of invariants $\{h_1, \dots, h_m\}$ using Algorithm 2.2.10.
3. Compute a Gröbner basis B_0 for the ideal generated by

$$(\{h_1 - y_1, \dots, h_m - y_m\}) \subset \mathbb{C}[x_1, \dots, x_n, y_1, \dots, y_m]$$

with respect to the lexicographic monomial order induced from $x_1 > \dots > x_n > y_1 > \dots > y_m$. (Omit the secondary invariant $\{1\}$ from this process.)

4. Compute $B_0' := B_0 \cap \mathbb{C}[y_1, \dots, y_m]$.
5. Output: B_0' , the Gröbner basis of the ideal J where $\mathcal{Z}(J) = \mathbb{C}^n/G \hookrightarrow \mathbb{C}^m$.

Recall from Definition 2.1.4 that sets can be invariant as well as elements. Let $\mathcal{F} = \{f_1, \dots, f_s\}$ be a set of polynomials which is invariant under the group action of G ; $\forall g \in G, \forall i, \exists j : f_i \circ g = f_j$. Then its ideal $I = (\mathcal{F})$ is a set invariant under the action of G on $\mathbb{C}[x_1, \dots, x_n]$, and its locus $\mathcal{Z}(\mathcal{F}) = \mathcal{Z}(I)$ is a set invariant under the action of G on \mathbb{C}^n . The problem that arises when applying Gröbner bases to obtain $\mathcal{Z}(I)$ from I highlights the chief failure of Gröbner bases in Invariant Theory: *Gröbner bases do not preserve symmetry*. If we did apply a Gröbner basis method to obtain $\mathcal{Z}(I)$ from I , we start with a symmetric set, turn it into an asymmetric set, the Gröbner basis, and we are faced with the task of restoring the symmetry, by using the asymmetric set to compute the symmetric variety $\mathcal{Z}(I)$.

We want some sort of way to preserve the symmetries in our Gröbner basis calculations, a method that produces an invariant Gröbner basis. To this end, we introduce the following term.

Definition 3.5.4. Since the variety $\mathcal{Z}(I) \subset \mathbb{C}^m$ is invariant under the action of G , we can define the **relative orbit variety** $\mathcal{Z}(I)/G$, whose points are the G -orbits of zeroes of I .

It can be shown that $\mathcal{Z}(I)/G$ is an algebraic subvariety of \mathbb{C}^n/G , and hence an algebraic subvariety of \mathbb{C}^m . In order to preserve the symmetry during Gröbner calculations, we compute a Gröbner basis for the relative orbit variety $\mathcal{Z}(I)/G$ rather than for $\mathcal{Z}(I)$ itself, and reconstruct the properties of $\mathcal{Z}(I)$ from $\mathcal{Z}(I)/G$.

Algorithm 3.5.5. (Computing the relative orbit variety)

1. Input: monomial ordering and B_0 from Algorithm preprocess.
2. Compute Gröbner basis B_1 , using the set $\{\mathcal{F} \cup B_0\}$.
3. Compute $B_2 := B_1 \cap \mathbb{C}[y_1, \dots, y_n]$.
4. Output: B_2 , the Gröbner basis for the ideal of $\mathcal{Z}(I)/G$.

The process for working "backwards" from $\mathcal{Z}(I)/G$ to examine $\mathcal{Z}(I)$ works as follows. Each point $\tilde{y} \in \mathcal{Z}(I)/G$ gives rise to a unique G -orbit in $\mathcal{Z}(I)$. (The tilde used here should not be confused with the tilde operation in Section 3.2, where it denoted the k -algebra homomorphism associated with a morphism, not specific coordinates in a relative orbit variety.) Such an orbit is a subset of \mathbb{C}^n of cardinality $\leq |G|$. The points in the orbit corresponding to \tilde{y} can be computed by substituting the coordinates of $\tilde{y} = (\tilde{y}_1, \dots, \tilde{y}_m) \in \mathbb{C}^m$ for the variables y_1, \dots, y_m in B_0 , and then the desired orbit equals the subvariety of \mathbb{C}^m which is defined by the Gröbner basis $B_0^*(\tilde{y}) \subset \mathbb{C}[x_1, \dots, x_n]$.

Finally, we obtain an invariant Gröbner basis by applying the mapping $y_i \mapsto h_i(x_1, \dots, x_n)$, $1 \leq i \leq m$ to B_2 . The result, B_2' , is invariant under the group action, and a Gröbner basis for the ideal (\mathcal{F}) , although it is not necessarily a reduced Gröbner basis.

Example 3.5.6. Applying Variety Algorithms 1

Let $n = 2$ and consider the set of polynomials $\mathcal{F}_1 = \{f_1, f_2\} \subset \mathbb{C}[x_1, x_2]$ where

$$\begin{aligned} f_1 &= x_1x_2 - 1 \\ f_2 &= x_1^2 + x_2^2 - 4 \end{aligned}$$

Figure 3.1: Intersection of Circle and Hyperbola

We want to find the locus associated by the ideal $I_1 = (\mathcal{F}_1)$. This set was specially chosen, since the solution is the intersection of the equations $x_1x_2 = 1$ and $x_1^2 + x_2^2 = 4$, which represent figures we can easily graph. f_1 is a hyperbola and f_2 is a circle with radius 2 centered at the origin (as shown in 3.1).

In this case, we can easily tell that $\mathcal{Z}(I_1)$ consists of 4 points, but we can easily determine it computationally as well. The Gröbner basis for I_1 with respect to the lexicographic ordering $x_1 > x_2$ is $B = \{\underline{x_1} + x_2^3 - 4x_2, \underline{x_2^4} - 4x_2^2 + 1\}$. Using the underlined leading terms, we can calculate the set of standard monomials : $B^* = \{1, x_2, x_2^2, x_2^3\}$. Using Theorem 3.4.9, B^* is a \mathbb{C} -vector space basis for the quotient ring $\mathbb{C}[x_1, x_2]/I_1$. Thus, $\mathbb{C}[x_1, x_2]/I_1$ is a \mathbb{C} -vector space of dimension 4, which further implies that the variety $\mathcal{Z}(I_1)$ consists of 4 points in affine 2-space, possibly counting multiplicities. We can calculate them explicitly using the methods discussed in Section 1.5:

$$\begin{aligned} &((\sqrt{6}/2 - \sqrt{2}/2), (-\sqrt{6}/2 - \sqrt{2}/2)), \quad (1) \\ &((\sqrt{6}/2 - \sqrt{2}/2), (\sqrt{6}/2 + \sqrt{2}/2)), \quad (2) \\ &((\sqrt{6}/2 + \sqrt{2}/2), (-\sqrt{6}/2 - \sqrt{2}/2)), \quad (3) \\ &((-\sqrt{6}/2 - \sqrt{2}/2), (-\sqrt{6}/2 + \sqrt{2}/2)). \quad (4) \end{aligned}$$

So far, we haven't done anything we didn't already know how to do. Now, we will apply Algorithm 3.5.5 to analysis the orbit space of $\mathcal{Z}(I_1)$. The set \mathcal{F}_1 is invariant under the group action of the permutation group S_2 . This group action acting on $\mathbb{C}[x_1, x_2]$ has the invariant subring $\mathbb{C}[x_1, x_2]^{S_2}$, which we know to be generated by $x_1 + x_2$ and $x_1^2 + x_2^2$ by our work with the symmetry groups in Section 2.1. Using this set of invariants, we apply the preprocessing of Algorithm 3.5.3. This results in $B_0 = \{2x_2^2 - y^2 + x_2y_1 + y_1^2, x_1 + x_2 - y_1\}$. Next, we calculate the Gröbner basis of the set $\{2x_2^2 + x_2y_1 + y_1^2, x_1 + x_2 - y_1, x_1x_2 - 1, x_1^2 + x_2^2 - 4\}$, which gives us $B_1 = \{y_2 - 4, y_1^2 - 6, x_2^2 - x_2y_1 + 1, x_1 + x_2 - y_1\}$. Finally, we take the intersection of B_1 with $\mathbb{C}[y_1, y_2]$, which gives us $B_2 = \{\overline{y_2} - 4, \overline{y_1^2} - 6\}$. Taking the set of standard monomials, $B_2^* = \{1, y_1\}$, and so the orbit variety $\mathcal{Z}(I_1)/S_2$ has at most two points. (This is plainly evident just by looking at B_2 , but it's nice to get the confirmation.) Specifically,

$$\begin{aligned} \mathcal{Z}(I_1)/S_2 = \quad & \{(\sqrt{6}, 4), \quad (a) \\ & (-\sqrt{6}, 4)\}, \quad (b) \end{aligned}$$

a result we can get quite simply by applying the methods in Section 1.5.

We can find which points in $\mathcal{Z}(I_1)/S_2$ correspond to which orbits of points in $\mathcal{Z}(I_1)$ by substituting the points of $\mathcal{Z}(I_1)/S_2$ into B_0 and finding the matching points in the resulting subvariety. Substituting (a) into B_0 yields $B_{0,a} = \{2x_2^2 + \sqrt{6}x_2 + 6 - 4, x_1 + x_2 + \sqrt{6}\}$. The solutions to this set are points (1) and (4), so the point (a) $\in \mathcal{Z}(I_1)/S_2$ corresponds to the orbit $\{(1), (4)\} \subset \mathcal{Z}(I_1)$. Similar calculations show that the point (b) corresponds to the orbit $\{(2), (3)\}$. This matches exactly what we get if we calculate the orbits directly using the action of S_2 on the points (1) and (2).

Last, we can obtain an invariant Gröbner basis for the variety $\mathcal{Z}(I_1)$ by using the transformation $y_i \mapsto h_i(x_1, x_2)$, where $h_1 = (x_1 + x_2)$, $h_2 = (x_1^2 + x_2^2)$ to B_2 . The result is $B_2' = \{x_1^2 + x_2^2 - 4, x_1^2 + 2x_1x_2 + x_2^2 - 6\}$. (Note that this is not a reduced Gröbner basis.)

Example 3.5.7. Applying Variety Algorithms 2

Let $n = 3$ and consider the set of polynomials $\mathcal{F}_2 = \{f_1, f_2, f_3\} \subset \mathbb{C}[x_1, x_2, x_3]$ where

$$\begin{aligned} f_1 &= x_1^2 + x_2^2 + x_3^2 - 4 \\ f_2 &= x_1x_2x_3 - 1 \\ f_3 &= x_1 + x_2 + x_3 \end{aligned}$$

The Gröbner basis of the ideal $I_2 = (\mathcal{F}_2)$ with respect to the lexico-

Figure 3.2: 3-D Representation of f_1, f_2, f_3 .

graphic order induced from $x_1 > x_2 > x_3$ equals $B = \{\overline{x_3^3} - 2x_3, \overline{x_2^2} + x_3^2 - 2 + x_2x_3, \overline{x_1} + x_2 + x_3\}$. The corresponding set of standard monomials is $\{1, x_2, x_3, x_2x_3, x_3^2, x_2x_3^2\}$, which implies that the variety $\mathcal{Z}(I_2)$ consists of 6 points in affine 3-space, possibly counting multiplicities. Geometrically, the equations represent figures like in 3.2, which means $\mathcal{Z}(I_2)$ is the intersection of a sphere, a plane, and a 3-D analog of hyperbola. Applying the methods in Section 1.5 shows that the points are:

$$(-1, 1/2 + \sqrt{5}/2, 1/2 - \sqrt{5}/2), \quad (1)$$

$$(1/2 + \sqrt{5}/2, -1, 1/2 - \sqrt{5}/2), \quad (2)$$

$$(-1, 1/2 - \sqrt{5}/2, 1/2 + \sqrt{5}/2), \quad (3)$$

$$(1/2 - \sqrt{5}/2, -1, 1/2 + \sqrt{5}/2), \quad (4)$$

$$(1/2 - \sqrt{5}/2, 1/2 + \sqrt{5}/2, -1), \quad (5)$$

$$(1/2 + \sqrt{5}/2, 1/2 - \sqrt{5}/2, -1). \quad (6)$$

The set \mathcal{F}_2 is invariant with respect to the symmetric group S_3 of 3×3 -

permutation matrices. The invariant ring $\mathbb{C}[x_1, x_2, x_3]^{S_3}$ is generated by

$$\begin{aligned} b_1 &= x_1 + x_2 + x_3, \\ b_2 &= x_1^2 + x_2^2 + x_3^2, \\ b_3 &= x_1^3 + x_2^3 + x_3^3. \end{aligned}$$

Applying the Grobner basis calculations results in:

$$\begin{aligned} B_0 &= \{-2y_3 + 3x_3y_1^2 - y_1^3 + 6x_3^3 - 3x_3y_2 - 6x_3^2y_1 + 3y_1y_2, 2x_2^2 + 2x_3^2 - y_2 + 2x_2x_3 - 2x_2y_1 - 2x_3y_1 + y_1^2, x_1 + x_2 + x_3 - y_1\}, \\ B_1 &= \{-3 + y_3, -4 + y_2, y_1, -1 + x_3^3 - 2x_3, x_2^2 + x_3^2 - 2 + x_2x_3, x_1 + x_2 + x_3\}, \\ B_2 &= \{-3 + \overline{y_3}, -4 + \overline{y_2}, \overline{y_1}\}, \\ B_2^* &= \{1\}. \end{aligned}$$

Thus, $\mathcal{Z}(I_2)/S_3$ consists of a single point, $(0, 4, 3)$, which means $\mathcal{Z}(I_2)$ has only the single orbit consisting of $\{(1), (2), (3), (4), (5), (6)\}$. This result matches what we get if we compute the orbit of of any element in $\mathcal{Z}(I_2)$.

The invariant Gröbner basis for $\mathcal{Z}(I_2)$ is $B_2' = \{x_1^3 + x_2^3 + x_3^3 - 3, x_1^2 + x_2^2 + x_3^2 - 4, x_1 + x_2 + x_3\}$.

Example 3.5.8. Applying Variety Algorithms 3

Again, let $n = 3$, and let the fundamental invariants and the group action be the same as in Example 3.5.7. This time, however, let $\mathcal{F}_3 = \{f_1, f_2\} \subset \mathbb{C}[x_1, x_2, x_3]$ where

$$\begin{aligned} f_1 &= x_1^2 + x_2^2 + x_3^2 - 4 \\ f_2 &= x_1 + x_2 + x_3 \end{aligned}$$

This set still contains the sphere and the plane, but we remove the third equation. From Figure 3.2, it is clear that removing the third equation results in an infinite set of points for $\mathcal{Z}(I_3)$, where $I_3 = (\mathcal{F}_3)$. We can also show this computationally, as the Gröbner basis is $B = \{x_2^2 + x_3^2 - 2 + x_2x_3, \overline{x_1} + x_2 + x_3\}$, and the associated set of standard monomials is $B^* = \{1, x_2, x_3, x_3x_2, x_3^2, \dots\}$. B^* is an infinite set, which means that $\mathcal{Z}(I_3)$ is an infinite set as well. Every element of $\mathcal{Z}(I_3)$ is of the form

$$((1/2)x \pm 1/2(-3x^2 + 8)^{1/2}, -(1/2)x \pm 1/2(-3x^2 + 8)^{1/2}, x), \quad x \in \mathbb{C}.$$

For example, $(\sqrt{2}, -\sqrt{2}, 0) \in \mathcal{Z}(I)$, when we choose $x = 0$. Also, note that the $\mathcal{Z}(I_2) \subset \mathcal{Z}(I_3)$.

Calculating the various Gröbner bases yields:

$$\begin{aligned}
B_0 &= \{-2y_3 + 3x_3y_1^2 - y_1^3 + 6x_3^3 - 3x_3y_2 - 6x_3^2y_1 + 3y_1y_2, 2x_2^2 + 2x_3^2 \\
&\quad - y_2 + 2x_2x_3 - 2x_2y_1 - 2x_3y_1 + y_1^2, x_1 + x_2 + x_3 - y_1\}, \\
B_1 &= \{-4 + y_2, y_1, 3x_3^3 - 6x_3 - y_3, x_2^2 + x_3^2 + x_2 * x_3 - 2, x_1 + x_2 + x_3\}, \\
B_2 &= \{\overline{y_2} - 4, \overline{y_1}\}, \\
B_2^* &= \{1, y_3, y_3^2, y_3^3, \dots\}
\end{aligned}$$

B_2^* is an infinite set, so the relative orbit variety $\mathcal{Z}(I_3)/S_3$ is also infinite. This is to be expected; since we have a finite group, S_3 , acting on an infinite set, $\mathcal{Z}(I_3)$, it partitions $\mathcal{Z}(I_3)$ into infinitely many orbits.

As an exercise, let's pick a point in $\mathcal{Z}(I_3)/S_3$ and trace it back to its orbit. $(0, 4, 0) \in \mathcal{Z}(I_3)/S_3$, and it corresponds to the subvariety given by the altered basis $B_{0,(0,4,0)} = \{6x_3^3 - 12x_3, \overline{2x_2^2} + 2x_3^2 - 4, \overline{x_1} + x_2 + x_3\}$. Then $B_{0,(0,4,0)}^* = \{1, x_2, x_3, x_2x_3, x_3^2, x_2x_3^2\}$, which means the orbit has a maximum of six elements. This matches what we know about S_3 ; when we rearrange any 3-tuple (x_1, x_2, x_3) , there are six different outcomes if x_1, x_2, x_3 are distinct, and fewer outcomes if they are not. In particular, the solution set for $B_{0,(0,4,0)}$ is

$$(\sqrt{2}, -\sqrt{2}, 0), \quad (1)$$

$$(-\sqrt{2}, \sqrt{2}, 0), \quad (2)$$

$$(\sqrt{2}, 0, -\sqrt{2}), \quad (3)$$

$$(-\sqrt{2}, 0, \sqrt{2}), \quad (4)$$

$$(0, -\sqrt{2}, \sqrt{2}), \quad (5)$$

$$(0, \sqrt{2}, -\sqrt{2}), \quad (6)$$

And so, the point $(0, 4, 0) \in \mathcal{Z}(I_3)/G$ corresponds to the orbit $\{(1), (2), (3), (4), (5), (6)\} \subset \mathcal{Z}(I)$. Any other orbit can be calculated in a similar manner.

Finally, the invariant Gröbner basis for $\mathcal{Z}(I_3)$ is $B_2' = \{x_1 + x_2 + x_3, x_1^2 + x_2^2 + x_3^2 - 4\}$; exactly the same set as we started with.

Chapter 4

The Road Ahead

As this report draws to a close, we take a look at what has been done and what is yet to come. Unfortunately, we've only reached the point where we can properly understand the problem posed in the abstract; there is still some information lacking before it can be given the attention it deserves. However, we did come tantalizingly close. An overview of the Gröbner bases provided us with a variety of tools and algorithm to really get at the heart of the matter, and a long look at invariant theory ultimately culminated in a method for finding the invariants necessary to consider the matter explicitly. The final section on Algebraic Geometry brought us to the very brink, giving means to analyse with reasonable thoroughness the structure of an algebraic set and its orbit space. Unfortunately, a semi-algebraic set is a more complicated animal, and we never quite got to addressing it.

A number of venues remain to be pursued if this topic is ever returned to. As far as the paper itself goes, sections 2.3, 2.4, and the appendix section on Krull Dimension would all benefit from further refinement and clarification. Also, there is a corollary to Theorem 3.4.1 in [1] relating to the multiplicity of the elements in the locus. It involves localization, and getting to the point where it could be explained would go a ways to explaining some of the background information in the prime ideal algorithm. Another topic to pursue is rest of Section 2.6 in [13], which discusses how to determine the image space of a variety under a group action. The papers [12] and [3] suggest a means to bridge the gap between invariant theory and semi-algebraic sets. In general, [4] should probably be approached in a more thorough manner to gain a better understanding of semi-algebraic sets. Further computer programs, based on the algorithms in this paper and elsewhere, may provide further in-

sights into the fundamental invariant questions. Comments contained in [8] and elsewhere suggest that a full knowledge of representation theory would further studies considerably. Finally, it would be interesting to spend some time comparing the material in Section 3.5 with the material in Section 3.2 concerning morphisms, and investigating whether any sort of correspondence can be made.

4.1 Acknowledgements

A number of people contributed to this project in various ways, and it behooves me to make mention of them. First, thanks to everyone who I forget to thank explicitly. (That should cover most people sufficiently.) Thanks to Wolfgang Gröbner, for lending his name to what is quickly becoming my favorite mathematical tool. Thanks to NSERC and the Mathematics and Statistics Department of the University of Saskatchewan for providing a means of enabling and encouraging the research work of undergraduate students. Thanks to Erin Dramitzki, Kerry Sproule, and my parents for providing the means to transport my unwieldy chef-d'oeuvre, the poster "Gröbner Bases and Invariant Theory", to and from the university campus. Thanks to my fellow summer students for making my experience more enjoyable and being there to lend an ear whenever a theory needed talking through. Last, and by no standard of measurement the least, thanks to my supervisor, Professor Salma Kuhlmann. Thank you for providing the space to find my own path, for supplying the support when I needed help, and most of all, for giving me this opportunity to learn.

Appendix A

Appendix A

A.1 Invariant Algorithm

Note: All of the subalgorithms mentioned prior to the Invariant Algorithm are contained inside the invariant procedure below, and so I deemed it pointless to repeat them separately. The exception to this is the decomposition inclusion algorithm, which proved to be too difficult to program, because of the difficulty in programming its monomial ordering.

```
with(Maplets[Elements]): #necessary for input program
with(combinat):          #necessary for gener program
with(StringTools):      #necessary for gener program
with(Groebner):         #necessary for Groebner basis calculations
with(ListTools):        #necessary for Flatten command
with(PolynomialIdeals): #necessaryfor LeadingTerm command with
(LinearAlgebra): #necessary for random matrix
```

```
#This program is designed to calculate the primary and (hopefully)
# secondary invariants of an invariant ring,
#given the number of variables the full ring is made of (that is,
#given n in  $C[x_1, \dots, x_n]$ ) and given the effects each element of a
#finite group has on each variable. In order to do so, we divide
#the task into numerous smaller parts, like so:
# Master- the master program. manages the individual subsections,
```

```

#           and maintains the necessary parameters.
# Input-    takes the number of variables
#           from the user, as well as the number of elements in
#           the group, followed by how each element effects each
#           indeterminant. It returns all three sets of information.
# Primary-  master program for the primary generation. It calls the
#           program for generating and invariating the monomial,
#           calls up the program to test for containment in the set
#            $\text{rad}(\langle Q \rangle)$  of primary invariants, where  $Q$  is the set of
#           primary invariants, and calls up the program to test if
#            $\text{rad}(\langle Q \rangle)$  is  $M$ , # where  $M$  is the irrelevant ideal  $M =$ 
#            $\langle x_1, \dots, x_n \rangle$ . Finally, if necessary, it calls up the
#           program to modify  $Q$  into an algebraically independent
#           set  $P$  of invariants with  $\text{rad}(\langle P \rangle) = M$ . It takes all the
#           data from input and returns a set of primary invariants.
# Secondary- master program for the secondary generation. If it
#           actually worked, which it doesn't, it would keep track
#           of the set of secondary invariants  $S$ , and a bound of the
#           number of secondary invariants. It generates monomials,
#           and calls up a program to test whether the invariant
#           counterpart of the monomial lies in the  $C[P]$ -module
#           generated by  $S$ , and adds the invariant counterpart to  $S$ 
#           if it does not. It takes in the set of primary invariants,
#           and returns the set of secondary invariants.
# Output-   This program takes in the primary and
#           secondary invariants and prints out the Hironaka
#           decomposition.
# Radcont-  This program tests whether an invariant is contained
#           within the radical ideal generated by  $Q$ . It takes in
#            $m_i$  and  $Q$ , and returns either true or false.
# Irrel-    This program tests whether  $\text{rad}(\langle Q \rangle)$  is equivalent to the
#           irrelevant ideal  $M = \langle x_1, \dots, x_n \rangle$ . It takes in  $Q$ 
#           and returns either true or false.
# modQ-     This program takes a set of dependent invariants,  $Q$ , and
#           returns a set of independent invariants,  $P$ .
# testmod-  This program tests whether a given invariant lies in the
#            $C[P]$  module generated by  $S$ . It takes a set of
#           primary invariants and a set of secondary invariants and

```

```

#         returns true or false.
# gener-   This program generates a monomial and computes its invariant,
#         then continues to generate until it
#         produces a non-zero polynomial. It takes in a set of
#         conditions for the monomial degree, and returns the
#         degree of the monomial and its invariant equivalent.

Master:=proc();
Input();
Primary();
Secondary();
Output();
end proc;

Input:=proc()global gsize, varinum, vari,
effectList,variset,t;local count,input1, input2,
input3,countout,countin;
# This program asks the user to input the indeterminants, the order
# of the group being considered, and the effects each element of the
# group has on the variables. Then it turns the input into global
# variables, so every program can use them.
# (So much easier than sending them everywhere as parameters!)
input1:=runmaplet1();
variset:=parse(input1[1]); #This variable is the list of
#                               x1,...,xn variables
vari:=[variset]; #This variable is the xi variables, but in a list
#                               rather than a set
varinum:=nops(vari); #This variable represents the number of
#                               xi variables.
input2:=runmaplet2();
gsize:=parse(input2[1]); #This represents the size of the group G.
effectList:=[];
for count from 1 to gsize do
    input3:=runmaplet3(count);
    effectList:=[op(effectList),[parse(input3[1])]];
#     each element of effectList represents what that element does
#     to the set of indeterminates x1,...,xn.
end do;

```

```

end proc;

Primary:=proc()global Q, QequalsM, P, t; local
genresults,containment;
#This program covers a lot of bases. Basically, it fulfills the
#following algorithm:
#-1.    Fix a monomial order  $m_1 < m_2 < m_3 < m_4 < \dots$  which refines
#        the partial order given by total degree on the set
#        of monomials in  $C[X]$ .
#0.     Let  $t:=1$  and  $Q:= []$ ;
#1.     Repeat  $t:=t+1$  until  $mt^*$  is not in  $\text{rad}(\langle Q \rangle)$ .
#2.     Let  $Q := Q$  and  $\{mt^*\}$ . If  $\text{rad}(\langle Q \rangle)$  doesn't equal  $\langle x_1, \dots, x_n \rangle$ 
#        then goto 1.
#3.     If  $Q$  has cardinality  $n$ , then  $P = Q$ . Otherwise, modify  $Q$ 
#        to an algebraically independent set  $P$  of invariants with
#         $\text{Rad}(\langle P \rangle) = M$ .

QequalsM:=false;
while QequalsM = false do
    t:=t+1;
    genresults:=gener();
    if genresults <> 0 then
        containment:=Radcont(genresults);
        if containment = false then #Then we add the invariant to Q,
                                    #and check if  $\langle Q \rangle = \langle x_1, \dots, x_n \rangle$ .
            Q:=[op(Q),genresults];
            #print("Q is now",Q);
            QequalsM :=Irrel();
        end if;
    end if;
end do; print(Q);
#That takes care of steps 1) and 2). The third step is to turn Q into
# an independent set, if necessary.
    if nops(Q) <> varinum then
        modQ();
    else P:=Q;
    end if;
end proc;

```

```

Secondary:=proc();
#The Secondary procedure is currently unavailable, as I can't seem
#to get Maple to observe the proper monomial ordering necessary
#for the Grobner basis calculation.
end proc;

gener:= proc() global t; local startvector, stopgenerate, first,
monvec, mono, stringMono, result, countout, countin, sum, invari,
count, ylist, counteffect,countvari,checkvec;
# This program loops through all monomials starting at the point
# (start,0,...0) and ending when it finds a monomial of that
# total degree that is nonzero, or when there are no polynomials
# of that total degree left to test. It does so by creating a
# list consisting of [start,0,...0], then using the list's vectoint
# to determine the starting point of the monomial set, and
# continues, until the integer of the first element in the vector is
# greater than start (or until it finds a nonzero #invariant).
# The computation of the invariant goes as follows: it takes the
# monomial, replaces its xi's with yi's, and #replaces its yi's
# with the conditions on the indeterminants for a given element, and
# saves the result, for all elements. Then it adds all the results for
# the element, and divides the sum by gsize. If the result is nonzero,
# we have our invariant.

stopgenerate:=false;          #stopgenerate is the condition that
tells when the program ends #first keeps track of the
#number of the vector whose monomial we're testing while
stopgenerate = false do
  #print(t);
  monvec:=inttovec(t,varinum);  #monvec is the starting vector
  mono:=1;                      #mono is the monomial being tested.
  #print(monvec,t);
  for count from 1 to varinum do
    mono:=mono*vari[count]^(monvec[count]);
  end do;
  result:=effectList;

```

```

#print("We're testing the monomial",mono);
#Ok, we've run into a bit of a snag.  If a particular change is x1<->x2,
# my algorithm will convert the x1 into x2,
#Then convert the x2 back into x1.  To get around this, we change every
# xi variable into yi, then change yi into
#the appropriate transformation.  It's a bit convoluted, but it works.
ylist:=[];
for count from 1 to varinum do
ylist:=[op(ylist),y[count]];
end do;
for counteffect from 1 to nops(effectList) do
  result[counteffect]:=mono;
  for countvari from 1 to varinum do
    result[counteffect]:=subs(vari[countvari] = ylist[countvari],
      result[counteffect]);
  end do;
  for countvari from 1 to varinum do
    result[counteffect]:=subs(ylist[countvari]=effectList[counteffect]
      [countvari],result[counteffect]);
  end do;
  #print(mono,"becomes",result[counteffect], "under transformation",
  #  effectList[counteffect]);
end do;
#print(mono,result);
sum:=0; # sum represents the sum of all terms in result, divided by the
#  order of the group.
for count from 1 to nops(result) do
  sum:=sum+expand(result[count]);
end do;
sum:=sum/gsize;
#print(mono,"is taken by the Reynolds operator to",sum);
if sum <> 0 then
  sum:=1/LeadingCoefficient(sum,plex(variset))*sum;
  invari:=sum;      #invari is the invariant that the gener
                    #program returns.
  stopgenerate:=true;
else
  t:=t+1;

```

```

    end if;
end do;
invari;
end proc;

```

```

Radcont:=proc(given)local basis1, test;
# This program determines whether a given monomial is contained in
# an ideal rad(<Q>), or not, and returns true or false,
# respectively. It does so by calculating the Grobner basis of the
# current set of invariants, along with the additional
# monomial to be tested for containment,times a new variable, minus
# one. The result is that the monomial is contained
# in rad(<Q>) iff basis1 contains 1.
basis1:=gbasis(Flatten([Q,given*z-1]),plex(variset,z));
if basis1[1] = 1 then
    test:=true;
    #print(given,"is contained in the radical");
else
    test:=false;
    #print(given, "is not contained in the radical");
end if;
test;
end proc;

```

```

Irrel:=proc()local basis,stringbasis,varistring,count1,count,count2,
QequalsMTest,man,man2,soltest;
#This program determines whether rad<Q> = M = <x1,x2,...,xn>,
#which holds iff there is no nontrivial solution to the set of
#homogenous polynomials that make up Q, which holds iff a monomial
#of the form xi ^ (ji) occurs among the initial monomials in G for
#every i, i between 1 and n. What the program does is calculate
#the Grobner basis of Q, turn the list of leading terms into a
#string list, search the string for a given variable, and search
#for whether the leading term containing #the variable has any
#other terms. If it doesn't, a true value is stored for that
#variable. If it does, the program searches through the other
#leading terms for one containing just a power of xi. If no such
#power is found, a false value is stored. The same is done

```

```

#for every variable. At the end, if all variables have a true
#value associated with them, the #program returns true; otherwise, it
#returns false.
#print("Inside the Irrel procedure");
basis:=gbasis(Q,plex(variset)); #This variable keeps track of the
#                               basis of the set
        #print(basis);
stringbasis:=basis;      #These variables are the same as vari and
#                               basis, but as strings, not polynomials.
varistring:=vari;
for count1 from 1 to nops(basis) do
    stringbasis[count1]:=convert(LeadingMonomial(basis[count1]
        ,plex(variset)),string);
    #print(basis[count1]);
end do;
#print (stringbasis,basis); for count1 from 1 to varinum
do
    varistring[count1]:=convert(vari[count1],string);
end do;
#print(vari,varistring);
QequalsMTest:=vari;
#This stores the truth values for the test; true represents xi^ji
#exists.
for count from 1 to nops(QequalsMTest) do
    QequalsMTest[count]:=false;
    #We set the default position as false, assuming Q does not equal M.
end do;
for count1 from 1 to varinum do for count2 from 1 to nops(basis) do
    man:=Search(varistring[count1], stringbasis[count2]);
    #If it finds the variable, it searches for any other variables
    # in that monomial.
    if man <> 0 then
        #print("The term",stringbasis[count2],"contains",
            #vari[count1]);
        man2:=Search("*",stringbasis[count2]);
        if man2 <> 0 then #Then the term contains other variables
            # as well and we dismiss it
            #print("But it also contains another variable, so

```

```

        # it's inelligible");
        else
        #If there are no other variables in that monomial, then
QequalsMTest is true for that variable
            QequalsMTest[count1]:=true;
            end if;
        #else print("The term",stringbasis[count2],"does not contain"
        #,vari[count1]);
        end if;
end do; #If QequalsMTest is still false at this point, then there
# is no variable of the form xi^ji, and we end the search.
if QequalsMTest[count1] = false then
    #print("There is no term of the form",vari[count1],"^ji");
end if;
end do; # if any term in the QequalsMTest is false, then rad<Q>
# does not equal M; we check for that condition here.
soltest:=true;
for count from 1 to nops(QequalsMTest) do
    if QequalsMTest[count] = false then
        soltest:=false;
    end if;
end do;
#soltest determines whether any variable failed to be
#represented as xi^ji in the basis. If it is false, then some
#variable failed.
#if soltest then
    print("rad<Q> = M");
else
print ("rad<Q> does not equal M"); end if; soltest; end proc;

modQ:=proc();
#This procedure is done to check for algebraic
#dependence. What it does is take the set, calculate the Grobner
#basis of #{f1-y1,...,fm-ym} wrt plex. If G', the intersection of
#G and C[y] = empty set, then Q is algebraically independent.
#Otherwise, the elements of G' make up the dependence, replacing
#y_i with f_i. poly2:=Q;
for counter from 1 to nops(Q)do

```

```

    poly2[counter]:=poly2[counter]-z[counter];
end do;
#print("Consider the altered x-z set",poly2);
zorder:=z[1];
if nops(Q) > 1 then
    for counter from 2 to nops(Q) do
        zorder:=(zorder,z[counter]);
    end do;
end if;
basis:=gbasis(poly2,plex(variset,yorder));
#print("The altered x-z basis is",basis);
stringbasis:=basis;
zorder:=[zorder];
varistring:=vari;
for count1 from 1 to nops(basis) do
    stringbasis[count1]:=convert(basis[count1],string);
    #print(basis[count1]);
end do;
for count1 from 1 to nops(vari) do
    varistring[count1]:=convert(vari[count1],string);
end do;
dependencelist:=[];
for count1 from 1 to nops(basis) do
    #print("testing term",stringbasis[count1]);
    for count2 from 1 to nops(vari) do
        #print("testing variable", vari[count2]);
        man:=Search(varistring[count2], stringbasis[count1]);
        #print(man);
        if man <> 0 then
            #print("the variable",vari[count2],"is in",
                #stringbasis[count1]);
            count2:=nops(vari)+1;
        else
            # print("the variable",vari[count2],"is not in",
                # stringbasis[count1]);
        end if;
        if count2 =nops(vari) then
            dependencelist:=[op(dependencelist),

```

```

        stringbasis[count1]];
    end if;
end do;
end do;
if nops(dependencelist) = 0 then
    print("Set is algebraically independent");
else for count from 1 to nops(dependencelist) do
    for countvari from 1 to nops (yorder) do
        dependencelist[count]:=subs(yorder[countvari]
            = (Q[countvari]),dependencelist[count]);
    end do;
end do;
end if;
end proc;

#modQ:=proc(Q,variset);
#This procedure should take an algebraically
#dependent set Q and turns it into a algebraically independent set
#P. It does so by #taking d, the gcd of the degrees of the terms
#in Q, and generating an varinum x nops(Q) matrix such that the
#ideal #generated by p_i:= sum(from j = 1 to m) a_ij q_j^(d/d_j), i
#= 1,...,n has the irrelevant ideal M as its radical. We test
#this using the Irrel procedure. Then we test if the new set is
#of size varinum, and continue looping till it is. However,
it doesn't work, since the sum necessary involve radical exponents.
#First, we set up a vector for the degree of each monomial in Q:
#degreeelist:=Q;
#for count from 1 to nops(degreeelist) do
    # degreeelist[count]:=degree(Q[count],variset);
#end do;
#print(degreeelist,"degreeelist");
#The first thing we do is find d,
#which we do by looping through all the degreeelist values.
#d:=gcd(degreeelist[1],degreeelist[2]);
#These variables exist, because nops(Q) must be at least 2 to be
# a dependent set.
#if nops(Q) > 2 then
#   for count from 3 to nops(Q) do

```

```

#      d:=gcd(d,degreelist[count]);
#  end do; #end if;
#      #print("gcd",d);
#P:=Q;
#x:=varinum;
#print(nops(Q),varinum);
#while nops(P) <> x do
#  polylist:=vari;
#  matA:=RandomMatrix(varinum,nops(Q));
#  print(matA);
#  for i from 1 to varinum do
#    polylist[i]:=0;
#    for j from 1 to nops(Q) do
#      print(matA[i,j]);
#polylist[i]:=polylist[i]+matA[i,j]*simplify(Q[j]^(d/degreelist[j]));
#      #polylist[i] assuming real;
# print(polylist[i]);
#    end do;
#      print(polylist[i]);
#end do; #P:=polylist; #print(P);
#end do; #end proc;

#Output:=proc(); end proc;

runmaplet1:=proc() local mapletx,vari;
#This procedure asks the user for the variables.
mapletx := Maplet( Window( 'title'="4th", [
  "For C[x1,...xn], please enter the variables",
  TextField['TF1'](),
  [Button("OK", Shutdown(['TF1'])),
  Button("Clear", SetOption('TF1' = ""))]
] ) ): vari:=Maplets[Display]( mapletx );
vari;
end proc;

runmaplet2:=proc() local mapletx,groupnum;
#This program asks the
#user what the number of elements in the finite group is

```

```

mapletx :=Maplet( Window( 'title'="4th", [
    "Enter the number of elements in the group acting on C[X]",
    TextField['TF1'](),
    [Button("OK", Shutdown(['TF1'])),
    Button("Clear", SetOption('TF1' = ""))]
] ) ): groupnum:=Maplets[Display]( mapletx );
groupnum;
end proc;

runmaplet3:=proc(count) local mapletx,effectset;
#This program asks the user what effect a group element has on the
#variables of C[x1,..xn].
mapletx := Maplet( Window( 'title'=count, [
    "Enter for each variable what effect this element has on it
    (for example, if the element switches only x2 and x3,
    then type x1,x3,x2,...)",
    TextField['TF1'](),
    [Button("OK", Shutdown(['TF1'])),
    Button("Clear", SetOption('TF1' = ""))]
] ) ): effectset:=Maplets[Display]( mapletx );
effectset;
end proc;

```

A.2 Krull Dimension

While this program never connected itself directly to the material in the paper, it is elegant enough, in theory if not appearance, to warrant a place here.

Definition A.2.1. For any commutative ring R the **Krull dimension** of R is the maximum possible length of a chain $P_0 \subset P_1 \subset P_2 \subset \cdots \subset P_n$ of distinct prime ideals in R . The dimension of R is said to be infinite if R has arbitrarily long chains of distinct prime ideals.

[5] provides an algorithm for finding the Krull dimension of the quotient ring $\mathbb{C}[x_1, \dots, x_n]/I$, where the generators of I are given. The following is a program written in Maple, based on that algorithm.

```

with(LinearAlgebra):
with(Maplets[Elements]):
with(Groebner):
with(PolynomialIdeals):
with(StringTools):
Krull:=proc();
#The purpose of this program is to determine the Krull dimension of a ring
#R[x1,...,xn] / I. To do so, the user inputs a set of polynomials that
#generate a given ideal I, and the set of #variables the polynomials use.
#The program calculates the Grobner basis of the set. If there is a
#constant term in the basis then I = C[x], and the dimension, by convention,
# is -1. Otherwise, we find a subset M of the set of the variables
#of minimal cardinality such that for every nonzero g in G, the leading
#monomial LM(b) involves at least one variable from M. Then the dimension
# of I is n-|M|.
#1. First, we take in the user input, which consists of the polynomial set
# and the variable set.
variableset:=runmaplet1();
    #1.b ask user to input f1, ..., fm.
polyList:=runmapletb();
#2. Next, we convert the inputs into usable objects.
variableset:=parse(variableset[1]);
variablelist:=[variableset];
polyList:=[parse(polyList[1])];
#    print(variablelist);
#3. Next, we calculate the Grobner basis.
basis:=gbasis(polyList,tdeg(variableset));
print("The Grobner basis is",basis);
# if the basis is 1, then the dimension is -1, as mentioned, and we're done.
# Otherwise, we go on.
if basis[1] = 1 then
    finished:=true;
else
    finished:= false;
# Since we only need the leading terms, we create a list consisting only of
# them.
LTList:=basis;
for count from 1 to nops(basis) do

```

```

    LTList[count]:=LeadingTerm(basis[count],plex(variableset));
end do;
print("And its leading terms are",LTList);
#Now the hard part. The easiest thing to do is extract the variables used
#in the leading terms and take them as the set M. The problem with this
#approach is that it doesn't fulfill the minimum condition; if the
#leading terms were xy and x^2, for example, M = {x}, not {x,y}. A
#possible solution is to take one variable and check if it appears in
#the leading term set. If it does, we remove all other leading terms
#containing that variable from the LT set, and continue with the reduced
#set for the next variable. There's no guarantee that this method will
#get the minimal set either, but it will certainly get closer, and provides
#a minimum bound. Ideally, we could get the minimal set by ordering the
# variables so that we search for the one that appears the most. We'll
#see how computationally difficult that proves. First, we convert
#both sets into string form.
varistring:=variablelist;
Lstring:=LTList;
for count1 from 1 to nops(variablelist) do
    varistring[count1]:=convert(variablelist[count1],string);
end do;
for count from 1 to nops(LTList) do
    Lstring[count]:=convert(LTList[count],string);
end do;
#print(varistring, Lstring);
#Next, we loop through the Lstring set and count the number of appearances
# of each variable, storing it in the list
#countlist.
variright:=variablelist; #keeps track of the variables still left,
# but as polys, not strings
varileft:=varistring; #varistring keeps track of the variables still
# left appearing in the LTleft list.
countlist:=varileft; #countlist keeps track of the number of times the
# variable appears throughout the LTleft list.
LTleft:=Lstring; #LTleft keeps track of the Leading Terms containing
# variables not yet in M.
M:=[]; #M is the minimal set described above.
while nops(LTleft) > 0 do

```

```

countlist:=varileft;
for outcount from 1 to nops(varileft) do
    countlist[outcount]:= 0;
    for incount from 1 to nops(LTleft) do
        countlist[outcount]:=countlist[outcount]+ nops(
            [SearchAll(varileft[outcount],LTleft[incount])]);
    end do;
end do;
#print (countlist);
#Next, we search for the position of the biggest number in
#the countlist list, and take this to be our first variable.
max:=countlist[1];
for count from 1 to nops(countlist) do
    if countlist[count] > countlist[1] then
        max:= countlist[count];
    end if;
end do;
for count from 1 to nops(countlist) do
    if countlist[count] = max then
        store:=count; count:=nops(countlist)+1;
    end if;
end do;
vari:=varileft[store];
#print(vari);
#We add the variable to the set M
M:= [op(M), parse(vari)];
#print (M);
#Now begins the next hard task; removing the variable from
#varileft and variright, and removing the leading terms
#containing it in LTleft. First, the variright.
#print(variright);
temp:=[];
for count from 1 to nops(variright) do
    if variright[count] <> parse(vari) then
        temp:=[op(temp),variright[count]];
    end if;
    #print (temp);
end do;

```

```

    variright:=temp;
    #print(variright);
    temp:=[];
    for count from 1 to nops(varileft) do
        if varileft[count] <> vari then
            temp:=[op(temp),varileft[count]];
        end if;
    end do;
    varileft:=temp;
    temp:=[];
    for count from 1 to nops(LTleft) do
        man:=Search(vari, LTleft[count]);
        #print("TC",vari, LTleft[count],man);
        if man = 0 then
            temp:=[op(temp),LTleft[count]];
        end if;
    end do;
    LTleft:=temp;
    #print(variright, varileft, LTleft);
end do;
print("The set M is",M);
dim:= nops(variablelist)-nops(M);
#Now, after that masterful piece of programming, we print off the result.
end if;
if finished then
    print(" I = C",[variableset]," and its dimension is -1, by convention");
else
    print ("The dimension of the ideal is", dim);
end if;
end proc;

runmaplet1:=proc() local mapletx;
#This program asks the user what variables they want to use.
mapletx := Maplet( Window( 'title'="4th", [
    "Enter list of variables polynomials are expressed in
    (please use x1, x2, etc):",
    TextField['TF1'](),
    [Button("OK", Shutdown(['TF1'])), Button("Clear",

```

```

        SetOption('TF1' = ""))]
] ) ):
polyset:=Maplets[Display]( mapletx );
polyset;
end proc;
runmapletb:=proc() local mapletx;
#This program asks the user what the set of polynomials that generate I is
mapletx := Maplet( Window( 'title'="4th", [
    "Enter set of homogeneous polynomials:",
    TextField['TF1'](),
    [Button("OK", Shutdown(['TF1'])), Button("Clear",
        SetOption('TF1' = ""))]
] ) ):
polyset:=Maplets[Display]( mapletx );
polyset;
end proc;

```

Bibliography

- [1] Sugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer, 2003.
- [2] Jacek Bochnak, Michel Coste, and Marie-Françoise Roy. *Real Algebraic Geometry*, volume 36 of *A Series of Modern Surveys in Mathematics*. Springer, 1998.
- [3] L. Brocker. On symmetric semialgebraic sets and orbit spaces. *Journal of Pure and Applied Algebra*, 44:37–50, 1998.
- [4] Michel Coste. *An Introduction to Semialgebraic Geometry*.
- [5] Harm Derksen and Gergor Kemper. *Computational Invariant Theory*, volume 131 of *Encyclopedia of Mathematical Sciences*. Springer, 2002.
- [6] David S. Dummitt and Richard M. Foote. *Abstract Algebra*. John Wiley & Sons, Inc, third edition, 2004.
- [7] Karin Gatermann. *Computer Algebra Methods for Equivariant Dynamical Systems*, volume 1728 of *Lecture Notes in Mathematics*. Springer, 2000.
- [8] Karin Gatermann and Pablo Parrilo. Symmetry groups, semidefinite programs, and sums of squares. *Journal of Pure and Applied Algebra*, 194:95–128, 2004.
- [9] I. Martin Isaacs. Character Theory. In *The Concise Handbook of Algebra*, chapter E.2, pages 386–389. 2002.

- [10] Salma Kuhlmann. The Invariant Moment Problem. Lecture presented during the Conference on Positive Polynomials, Luminy March 14-18 2005.
- [11] Peter J. Olver. *Classical Invariant Theory*, volume 44 of *London Mathematical Society student texts*. Cambridge University Press, 1999.
- [12] G. Sartori and G. Valente. Tools in the orbit space approach to the study of invariant functions: rational parametrization of strata. *Journal of Physics A: Mathematical and General*, 36:1913–1929, 2003.
- [13] Bernd Sturmfels. *Algorithms in Invariant Theory*. Springer-Verlag Wein New York, 1993.