UCLA
UCLA Electronic Theses and Dissertations

Title
Sums-of-Squares Formulas over Arbitrary Fields

Permalink
https://escholarship.org/uc/item/9v03g1kh

Author
Lynn, Melissa Kathryn

Publication Date
2016

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

University of California
Los Angeles

Sums-of-Squares Formulas over Arbitrary Fields

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mathematics

by

Melissa Kathryn Lynn

2016

© Copyright by
Melissa Kathryn Lynn
2016

Abstract of the Dissertation

Sums-of-Squares Formulas over Arbitrary Fields

by

Melissa Kathryn Lynn

Doctor of Philosophy in Mathematics

University of California, Los Angeles, 2016

Professor Christian Haesemeyer, Chair

This dissertation is on sums-of-squares formulas, focusing on whether existence of a formula depends on the base field and on methods for finding sums-of-squares formulas.

We introduce a new approach to the study of sums-of-squares formulas, showing that a formula can be regarded as a solution to a system of polynomial equations and thus we can introduce the variety of sums-of-squares formula. This new perspective allows us to utilize tools of algebraic geometry in the study of sums-of-squares formulas, as well as raising new questions about the properties of this variety.

We use number theory and computational algebraic geometry to consider the question of whether existence of sums-of-squares formulas depends on the base field. We are able to show independence in some cases, however, the general case remains an open question. Furthermore, we show that existence of a formula over an algebraically closed field is computable.

We introduce an algebraic group action on the variety of sums-of-squares formulas, which gives us another way to study the structure of this variety and raises new questions about the structure of the action.

Finally, we provide algorithms for finding sums-of-squares formulas over the integers and finite fields. These algorithms use previous results about formulas over the integers, as well as the group action on the scheme of sums-of-squares formulas.

The dissertation of Melissa Kathryn Lynn is approved.

Burt Totaro

Alexander Sergee Merkurjev

Eric D'Hoker

Christian Haesemeyer, Committee Chair

University of California, Los Angeles

2016

To my parents . . .

for their constant love and support

Table of Contents

Acknowledgments

First and foremost, I would like to thank my advisor, Christian Haesemeyer, for his support and guidance. He found a project that is well suited to my strengths and interests, and provided direction and insight throughout my work on this topic. Without him, this dissertation certainly would not exist.

I would also like to thank the rest of my committee: Eric D'Hoker, Alexander Merkurjev, and Burt Totaro, for their consideration and input.

I'm very grateful for the support and love of my family through graduate school and the rest of my life. When I was in second grade, I told my parents that I hated math because I wasn't as quick at arithmetic as the other kids. Fortunately, they didn't accept this, and they encouraged me to take on new and interesting mathematical challenges throughout my education. Even though my mom thinks "functor" is a funny word and my dad knows "the functors of the group theory are really a very important part of it," they've always been willing to listen and show (feign?) interest in my work.

My friends at UCLA have played a huge role in making my graduate school experience an enjoyable one. I especially would like to thank Josie Bailey and Stephanie Lewkiewicz for being wonderful friends to me for the past few years. I would like to thank my favorite office mates and fellow bagel day sponsors, Sudesh Kalyanswamy and Zach Norwood. These are in addition to the rest of the friends I've made at UCLA: there are so many of you that I'm indebted to.

I'm also grateful for the support from my non-math friends. I would particularly like to mention Kaitlin Yoder-Henley, Lauren Seidel, and Henry Chan. Kaitlin, my co-pirate since 5th grade, followed me to Los Angeles and moved in a few blocks away from me. She might claim it was for work, but I know better. Lauren, my fellow 2nd-grade horse enthusiast, who regularly asks me when I'm moving back to Minnesota. And Henry, chief potato and regular source of advice.

I would also like to thank everyone I've played ultimate frisbee with over the past few years, especially Midas. They have played a huge part in keeping me sane through graduate school.

I'm very grateful for the fellowship support that I have received while in graduate school. I have been supported through the Eugene V. Cota-Robles Fellowship and NSF Research and Training Group grant.

Vita

2008	Research Experience for Undergraduates, Department of Mathematics, University of Wisconsin-Eau Claire, Eau Claire, WI. Researched the action of the symmetric group on polynomials.
2008-2011	Administrator for the Young Scholars Program, Department of Mathematics, University of Chicago, Chicago, IL. Handled the organizational aspects of running YSP.
2008-2009	Reader, Department of Mathematics, University of Chicago, Chicago, IL. Graded for Honors Calculus.
2009-2010	Research Experience for Undergraduates, University of Chicago, Chicago, IL. Researched Galois categories and categories of probability spaces.
2009-2010	Junior Tutor, Department of Mathematics, University of Chicago, Chicago, IL. Ran problem sessions and graded homework for Elementary Functions and Calculus.
2010	B.A. (Mathematics with Honors), University of Chicago.
2011-2016	Teaching Assistant, Department of Mathematics, University of California Los Angeles, Los Angeles, CA.
2012	Instructor, Center for Excellence in Engineering and Diversity, University of California Los Angeles, Los Angeles, CA. Gave lectures and ran workshops on Calculus.
2013-2016	Graduate Research, Department of Mathematics, University of California Los Angeles, Los Angeles, CA. Researched the existence of sums-of-squares formulas over arbitrary fields in the area of algebraic geometry.

2015 Graduate Student Instructor, Department of Mathematics, University of California Los Angeles, Los Angeles, CA. Taught Differential and Integral Calculus

Publications

Sums-of-Squares Formulas over Algebraically Closed Fields. arxiv.org, arXiv:1510.05369 [math.AG], October 2015.

CHAPTER 1

Introduction

In this dissertation, we study the existence of sums-of-squares formulas over arbitrary fields. A sums-of-squares formula of type $[r, s, n]$ over a field F is an identity of the form

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots + y_s^2) = z_1^2 + \cdots z_n^2$$

where the z_i are bilinear expressions in the x_j and y_k. Throughout this dissertation, we consider only fields of characteristic not 2. Existence of sums-of-squares formulas over fields of characteristic 2 is trivial.

A sums-of-squares formula can be thought of as defining composition or multiplication of elements in F^r and F^s, landing in F^n. For example, multiplication of complex numbers yields a sums-of-squares formula of type $[2, 2, 2]$.

The main question one would like to answer is

- For what $[r, s, n]$ does a sums-of-squares formula of type $[r, s, n]$ exist over a field F?

This dissertation concerns itself primarily with the two questions:

- Does existence of a sums-of-squares formula depend on the base field F? Explicitly, if a sums-of-squares formula of type $[r, s, n]$ exists over a field F, does a sums-of-squares formula of type $[r, s, n]$ necessarily exist over any other field K?
- How can we find sums-of-squares formulas?

We are able to answer the first question in the affirmative in some cases, however the general case remains unknown. For the second question, we introduce algorithms to find formulas, however, there are major obstacles in efficiency.

Sums-of-squares formulas were originally studied in the context of real normed division algebras, going back to the sixteenth century. More modern study of sums-of-squares formula began with the proof of the Hurwitz-Radon Theorem in 1922 and 1923 [14] [16]. Since then, sums-of-squares formulas have been studied using linear algebra, algebraic topology, differential geometry, combinatorics, and modern cohomological methods. In Chapter 2, we review the history of the study of sums-of-squares formulas, focusing on results most relevant to our questions.

In Chapter 3, we introduce a new perspective on sums-of-squares formulas, which was previously unknown. By considering the conditions on the coefficients in a sums-of-squares formula, we obtain an equivalent formulation of a sums-of-squares formula as a solution to a system of polynomial equations. Thus, we are able to introduce the variety of sums-of-squares formulas. This new approach enables us to apply tools of algebraic geometry to the study of sums-of-squares formulas, and the rest of this dissertation focuses primarily on these applications. Considering the variety of sums-of-squares formulas also opens many new questions about the structure of this variety.

In Chapter 4, we begin to study sums-of-squares formulas by considering the variety of formulas. We show that, for algebraically closed fields, existence in characteristic 0 and characteristic p is equivalent for all but finitely many p. As a corollary, we show that if a sums-of-squares formula exists over any field, then a formula exists over some finite field. We use the zeta function to find a bound on the degree of this field. Thus, sums-of-squares formulas can theoretically be detected by conducting searches over finite fields, which are discussed in Chapter 7.

In Chapter 5, we improve our main result from Chapter 4, establishing a bound on the finitely many p where differences can occur. We do this using Gröbner bases and a careful analysis of the coefficients that appear throughout Buchberger's algorithm. We are able to do this in spite of the fact that the scheme of sums-of-squares formulas is defined using too many variables and polynomials for it to be practical to actually compute Gröbner basis. Although the bound for "large enough" that we provide is atrocious, it is likely that

this bound could be improved significantly. This result, combined with the results from Chapter 4, mean that existence of sums-of-squares formulas over algebraically closed fields is theoretically computable, though not practical.

In Chapter 6, we define an algebraic group action of a product of orthogonal groups on the variety of sums-of-squares formulas. By showing that the action of one of these orthogonal groups is free, we obtain a lower bound on the dimension of the variety of sums-of-squares formulas when it is nonempty. In particular, this dimension is positive. The existence of this group action provides a new way to study the structure of the variety of sums-of-squares formulas, as well as raising new questions about the structure of the action itself.

In Chapter 7, we explore methods for conducting computer searches for sums-of-squares formulas over the integers and finite fields. We are able to use the group action from Chapter 6 to make these searches more efficient. Using a search over finite fields, we produce a formula over $\mathbb{F}_3$ that does not lift to a formula over the integers (existence of such a formula was previously unknown). We then show that this formula is related to a formula over the integers: considering the group action from Chapter 6, this formula is in the same orbit as a formula that lifts to a formula over the integers.

Finally, in Chapter 8, we review the open questions raised throughout this dissertations and avenues for future work on the topic.

CHAPTER 2

Background and Motivation

In this chapter, we review some of the previous work on sums-of-squares formulas and discuss the motivation for studying them. After defining sums-of-squares formulas, we begin with the original purpose of sums-of-squares formulas in the study of real normed division algebras. This work led to the conclusion that the only real normed division algebras are the real numbers, the complex numbers, the quaternions, and the octonions, and led to the more general Hurwitz-Radon theorem. Hurwitz also posed the general question of existence of sums-of-squares formulas. In 2.3, we review some results which use topology and geometry to study the existence of sums-of-squares formulas over $\mathbb{R}$. We continue with results on sums-of-squares formulas which hold over fields other than $\mathbb{R}$. In very special cases, we have that existence is independent of the base field. In 2.5, we review results on the special case of sums-of-squares formulas over the integers. In this case, existence has been studied using combinatorics and consistently signed intercalate matrices. Next, we survey some results on formulas over arbitrary fields which match the theorems over $\mathbb{R}$ obtained using topology and geometry. Finally, we list some motivation for continuing the study of sums-of-squares formulas.

Much of the material in this chapter can be found in more detail in Shapiro's excellent book [17]. Shapiro also provides introductory online notes on sums-of-squares formulas [18, 19, 20]. The background provided here is by no means comprehensive; instead we provide a sampling of previous work, focusing on results which are most relevant to this dissertation.

2.1 Basics

A sums-of-squares formula of type $[r, s, n]$ over a field F (of characteristic not 2) is an identity of the form

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots + y_s^2) = z_1^2 + \cdots + z_n^2$$

where each z_i is a bilinear expression in the x's and y's over F. Over the real numbers, a sums-of-squares formula of type $[r, s, n]$ is equivalent to a bilinear map $f : \mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R}^n$ such that $\|f(x, y)\| = \|x\| \cdot \|y\|$. In general, a sums-of-squares formula can be thought of as a product or composition law $F^r \times F^s \to F^n$ such that the "norm" of the product is the product of the "norms."

For an example, multiplication of complex numbers comes from a sums-of-squares formula of type $[2, 2, 2]$ over $\mathbb{R}$:

$$z_1 = x_1 y_1 - x_2 y_2$$

$$z_2 = x_1 y_2 + x_2 y_1$$

Similarly, there are formulas corresponding to multiplication of quaternions and octonions.

Sums-of-squares formulas have been studied for their relationship with real normed division algebras, which have applications in quantum mechanics. These formulas continue to be of interest because of their connection to important problems in geometry and topology.

The main question one would like to answer about sums-of-squares formulas is:

Question 2.1.1. Over a field F, for what $[r, s, n]$ do such formulas exist?

2.2 Early Work

Sums-of-squares formulas were originally studied in the context of real normed division algebras: existence of a sums-of-squares formula of type $[n, n, n]$ over $\mathbb{R}$ is equivalent to the existence of a real normed division algebra of dimension n.

The formula for $n = 2$ has long been known, and can be interpreted as the "law of moduli" for the complex numbers. In 1748, Euler found the 4-square identity, and once Hamilton discovered the quaternions in 1843, this was interpreted as the law of moduli for the quaternions. In 1818, Degen found an 8-square identity, followed in 1843 and 1845 by the independent discovery of the octonions by Graves and Cayley (respectively). After the 1840's, several mathematicians tried to find a 16-square identity, eventually concluding that this was impossible, though proofs were incomplete.

By studying sums-of-squares formulas, Hurwitz was able to settle the question of existence of real normed division algebras in [13], showing that the only ones are the real numbers, complex numbers, quaternions, and octonions. Hurwitz's theorem has been applied to the study of vector fields on spheres and homotopy groups of classical groups, as well as to quantum mechanics through the classification of simple Jordan algebras.

Hurwitz's proof uses elementary linear algebra, by giving an equivalent formulation of sums-of-squares formulas through the Hurwitz Matrix Equations, which we now introduce.

Begin with a sums-of-squares formula

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots + y_s^2) = z_1^2 + \cdots + z_n^2$$

of type $[r, s, n]$ over a field F, with z_i bilinear in the x's and y's.

View $X = (x_1, ..., x_r)$, $Y = (y_1, ..., y_s)$, and $Z = (z_1, ..., z_n)$ as column vectors, so that we have, for example,

$$z_1^2 + \cdots z_n^2 = Z^T \cdot Z$$

The bilinearity condition on a sums-of-squares formulas then becomes that $Z = AY$, where A is an $n \times s$ matrix with entries that are linear in X. A sums-of-squares formula can then be written as

$$(x_1^2 + \cdots + x_r^2)Y^T \cdot Y = Y^T A^T AY$$

Then, since Y consists of indeterminates, we must have:

$$A^T \cdot A = (x_1^2 + \cdots + x_r^2)I_s$$

Writing $A = x_1A_1 + x_2A_2 + \cdots + x_rA_r$, where A_i is an $n \times s$ matrix with constant entries, we arrive at the Hurwitz Matrix Equations.

Theorem 2.2.1. ***(Hurwitz Matrix Equations)*** *Consider a system of $n \times s$ matrices $A_1, A_2, ..., A_r$ over F satisfying*

$$A_i^T \cdot A_i = I_s = I_s \text{ for } 1 \leq 1 \leq r$$
$$A_i^T \cdot A_j + A_j^T \cdot A_i = 0 \text{ for } 1 \leq i, j \leq r \text{ and } i \neq j$$

Existence of such a system of matrices is equivalent to the existence of a sums-of-squares formula of type $[r, s, n]$.

Now, consider the special case $s = n$. Then, taking $B_i = A_1^{-1}A_i$, we have the equivalent condition:

Theorem 2.2.2. *Existence of a sums-of-squares formula of type $[r, n, n]$ over F is equivalent to existence of a system of $n \times n$ matrices $B_2, ..., B_r$ over F satisfying:*

$$B_i^T = -B_i \text{ for } 2 \leq i \leq r$$
$$B_i^2 = -I_n \text{ for } 2 \leq i \leq r$$
$$B_iB_j = -B_jB_i \text{ for } i \neq j$$

Hurwitz showed that the 2^{r-1} matrices obtained as products of the B_i are linearly independent, showing that $2^{r-2} \leq n^2$, and when $r = n$, this gives us the "1,2,4,8 Theorem":

Theorem 2.2.3. *There is a sums-of-squares formula of type $[n, n, n]$ if and only if n is 1, 2, 4, or 8.*

Thus, the only real normed division algebras are the real numbers, the complex numbers, the quaternions, and the octonions.

In 1923 and 1922, Hurwitz [14] and Radon [16] independently generalized this result, showing that

Theorem 2.2.4. (Hurwitz-Radon Theorem) *A formula of size* $[r, n, n]$ *exists if and only if* $r \leq \rho(n)$.

Here, ρ *is the Hurwitz-Radon function, defined by: if* $n = 2^{4a+b}n_0$ *where* n_0 *is odd and* $0 \leq b \leq 3$, *then* $\rho(n) = 8a + 2^b$.

In the 1940's, the Hurwitz-Radon Theorem was generalized and proved in some new ways:

- Eckmann (1943) used representation theory of finite groups.
- Lee (1948) used representations of Clifford algebras.
- Albert (1942) generalized the 1,2,4,8 Theorem to quadratic forms over arbitrary fields.
- Dubisch (1946) used Clifford algebras to prove the Hurwitz-Radon Theorem for quadratic forms over $\mathbb{R}$.

This theorem remains true over any field of characteristic not 2.

In Hurwitz's paper [13], he also posed the general question:

Question 2.2.5. For what r, s, n does a sums-of-squares formula of type $[r, s, n]$ exist over a field F?

Hurwitz's characterization of sums-of-squares formulas using the Hurwitz Matrix Equations will be useful in Chapter 6, where we define a group action on the variety of sums-of-squares formulas.

2.3 Results from Topology

In this section, we review previous results about sums-of-squares formulas that have been obtained using topology.

The following characterization follows immediately from the definition of a sums-of-squares formula:

Proposition 2.3.1. *There is a sums-of-squares formula of size* $[r, s, n]$ *over* $\mathbb{R}$ *if and only if there is a normed bilinear map* $\mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R}^n$.

Here, a map $f : \mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R}^n$ *is* normed *if* $\|f(x, y)\| = \|x\|\|y\|$.

Since a normed map will necessarily be nonsingular (i.e., if $f(x, y) = 0$, then $x = 0$ or $y = 0$) we have the following consequence:

Proposition 2.3.2. *If there is a composition formula of size* $[r, s, n]$ *over* $\mathbb{R}$*, then there is a nonsingular bilinear map* $\mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R}^n$.

In 1941, Hopf [12] proved the following theorem, which rules out the existence of a sums-of-squares formula over $\mathbb{R}$ for some $[r, s, n]$.

Theorem 2.3.3. (Hopf's Theorem) *If there is a formula of type* $[r, s, n]$ *over* $\mathbb{R}$*, then* $\binom{n}{k}$ *is even for* $n - s < k < r$.

This is proved by showing that a nonsingular bilinear map induces a map on projective spaces, which in turn induces a map on singular mod 2 cohomology. The result follows by studying this map.

Another result uses K-theory to eliminate some possible sums-of-squares formulas. The work is due to Atiyah [2], though the application to sums-of-squares formulas was pointed out by Yuzvinsky [27].

Theorem 2.3.4. *If there is a formula of type* $[r, s, n]$ *over* $\mathbb{R}$*, then*

$$\binom{n}{i} \equiv 0 \pmod{2^{\phi(r-1)-i+1}}$$

for $n - s < i \leq \phi(r - 1)$.

This is proved by showing that a sums-of-squares formula yields

$$n \cdot \xi = s \cdot \epsilon \oplus \eta$$

for some $(n - s)$-plane bundle η over $\mathbb{P}^{r-1}$. Here ξ is the canonical bundle and ϵ the trivial bundle on $\mathbb{P}^{r-1}$.

These two results do not distinguish between normed and nonsingular maps. The following construction, due to Lam [15], enables one to show, in some cases, that a normed map cannot exist even though a nonsingular one does. In particular, this construction can be used to show that formulas of type $[16, 16, 23]$ do not exist, while there is a nonsingular map of that type.

Construction 2.3.5. If f is a normed pairing of size $[r, s, n]$, then we have a map

$$H : \mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R} \times \mathbb{R}^n$$

given by $H(x, y) = (|x|^2 - |y|^2, 2f(x, y))$. This restricts to a Hopf map

$$H : S^{r+s-1} \to S^n$$

For any point q in the image of H, $H^{-1}(q)$ is a great circle cut out by a linear subspace $W_q \subset \mathbb{R}^{r+s}$. The differential dH induces a nonsingular bilinear pairing of type $[k, r+s-k, n]$, where k is the dimension of W_q. Then we can use nonexistence of certain nonsingular maps to show nonexistence of the original normed map.

Hopf's Theorem and the result from K-theory have since been generalized to arbitrary fields; these generalizations are discussed in 2.6.

2.4 Results over Arbitrary Fields

In this section, we review some of the early results on sums-of-squares formulas over fields other than $\mathbb{R}$.

Recall that the Hurwitz-Radon Theorem remains true over any field (of characteristic not 2), and this theorem tells us that a sums-of-squares formula of type $[r, n, n]$ exists if and only if $r \leq \rho(n)$. In 1980, Adem proved the following theorem, also dealing with very special values of r, s, n [1].

Theorem 2.4.1. ***(Adem's Theorem)*** *Let F be any field of characteristic not 2. Suppose there is a sums-of-squares formula of type $[r, n-1, n]$ over F.*

- *If n is even, then there is a sums-of-squares formula of type $[r, n, n]$, so $r \leq \rho(n)$.*
- *If n is odd, then there is a sums-of-squares formula of type $[r, n-1, n-1]$, so $r \leq \rho(n-1)$.*

K.Y. Lam and T.Y. Lam provided some evidence for existence of sums-of-squares being independent of the base field in characteristic 0, proving the Lam-Lam Lemma. This can be found in [17].

Lemma 2.4.2. Lam-Lam Lemma *If there is a sums-of-squares formula of type $[r, s, n]$ over $\mathbb{C}$, then there is a nonsingular bilinear map of size $[r, s, n]$ over $\mathbb{R}$.*

Note that this does *not* say that the $\mathbb{C}$ case and the $\mathbb{R}$ case are equivalent.

From this result, we have the corollary:

Corollary 2.4.3. *If there is a formula of type $[r, s, n]$ over a field F of characteristic 0, then there is a nonsingular bilinear map of size $[r, s, n]$ over $\mathbb{R}$.*

Once again, this does not show that existence of sums-of-squares formula is independent of the base field in the characteristic 0, since it is possible that there is a nonsingular bilinear map over $\mathbb{R}$ but no sums-of-squares formula, as in 2.3.5. This does, however, provide some support for that conjecture.

2.5 Formulas over the Integers

In this section, we review past results on sums-of-squares formulas over the integers, for which the problem is greatly simplified.

Suppose we have a sums-of-squares formula over $\mathbb{Z}$, so

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots y_s^2) = z_1^2 + \cdots z_n^2$$

and write

$$z_k = \sum_{i,j} \alpha_{ijk} x_i y_j$$

Multiplying out and comparing coefficients, we have:

$$\sum_k \alpha_{ijk}^2 = 1$$

$$\sum_k \alpha_{ijk}\alpha_{ijk'} = 0$$

$$\sum_k \alpha_{ijk}\alpha_{ij'k} = 0$$

$$\sum_k \alpha_{ijk}\alpha_{ij'k'} + \alpha_{ijk'}\alpha_{ij'k} = 0$$

This is equivalent to having vectors

$$a_{ij} = (\alpha_{ij1}, ..., \alpha_{ijn})$$

such that

$$a_{ij} \cdot a_{ij} = 1$$

$$a_{ij} \cdot a_{ij'} = 0$$

$$a_{ij} \cdot a_{i'j} = 0$$

$$a_{ij} \cdot a_{i'j'} + a_{ij'} \cdot a_{i'j} = 0$$

We visualize this as a matrix of unit vectors, with rows and columns consisting of orthogonal vectors, and satisfying the additional condition:

$$a_{ij} \cdot a_{i'j'} + a_{ij'} \cdot a_{i'j} = 0$$

Since the coefficients α_{ijk} are all integers, the a_{ij} are all (plus or minus) basic unit vectors, so must be chosen from the finite set $\{\pm e_1, ..., \pm e_n\}$.

Ignoring signs for the moment, our matrix must satisfy:

- The a_{ij} along each row and column are distinct (hence orthogonal).
- If $a_{ij} = a_{i'j'}$, then $a_{i'j} = a_{ij'}$.

Let ϵ_{ij} be the sign of a_{ij}. Then we also must have:

$$\epsilon_{ij}\epsilon_{ij'}\epsilon_{i'j}\epsilon_{i'j'} = -1$$

whenever $a_{ij} = a_{i'j'}$.

This motivates the definition of a consistently signed intercalate matrix, which were introduced by Yuzvinsky [27]:

Definition 2.5.1. Suppose M is an $r \times s$ matrix with entries taken from a finite set of "colors." Let M_{ij} be the (i, j)-th entry of M.

M is *intercalate* if:

- The colors along each row (resp. column) are distinct.
- If $M_{ij} = M_{i'j'}$, then $M_{ij'} = M_{i'j}$. (Every 2×2 submatrix involves an even number of distinct colors.)

An intercalate matrix has *type* (r, s, n) if it is an $r \times s$ matrix with at most n colors.

An intercalate matrix M is *consistently signed* if there exist $\epsilon_{ij} = \pm 1$ such that

$$\epsilon_{ij}\epsilon_{ij'}\epsilon_{i'j}\epsilon_{i'j'} = -1$$

whenever $M_{ij} = M_{i'j'}$. (Every 2×2 submatrix with only two distinct colors must have an odd number of minus signs.)

Then we can study sums-of-squares formulas over the integers by studying consistently signed intercalate matrices:

Lemma 2.5.2. *There is a sums-of-squares formula of type $[r, s, n]$ over the integers if and only if there is a consistently signed intercalate matrix of type (r, s, n).*

For example, the consistently signed intercalate matrix

$$\begin{pmatrix} 1 & -2 & 3 & 4 & 5 \\ 2 & 1 & 4 & -3 & 6 \\ 3 & 4 & -1 & 2 & 7 \end{pmatrix}$$

(with colors $\{1, 2, 3, 4, 5, 6, 7\}$) corresponds to the sums-of-squares formula over $\mathbb{Z}$ given by

$$z_1 = x_1y_1 + x_2y_2 - x_3y_3$$

$$z_2 = x_2y_1 - x_1y_2 + x_3y_4$$

$$z_3 = x_1y_3 + x_3y_1 - x_2y_4$$

$$z_4 = x_1y_4 + x_2y_3 + x_3y_2$$

$$z_5 = x_1y_5$$

$$z_6 = x_2y_5$$

$$z_7 = x_3y_5$$

In [24, 25, 26], for $r, s \leq 16$, Yiu was able to determine the smallest n for which there is a formula of type $[r, s, n]$ over $\mathbb{Z}$ by studying intercalate matrices using combinatorial methods and combining this with previous results from topology. We give these values in the below table, letting $r *_{\mathbb{Z}} s$ denote the smallest n for which there is a formula of type $[r, s, n]$ over $\mathbb{Z}$.

$r *_{\mathbb{Z}} s$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	2	2	4	4	6	6	8	8	10	10	12	12	14	14	16	16
3	3	4	4	4	7	8	8	8	11	12	12	12	15	16	16	16
4	4	4	4	4	8	8	8	8	12	12	12	12	16	16	16	16
5	5	6	7	8	8	8	8	8	13	14	15	16	16	16	16	16
6	6	6	8	8	8	8	8	8	14	14	16	16	16	16	16	16
7	7	8	8	8	8	8	8	8	15	16	16	16	16	16	16	16
8	8	8	8	8	8	8	8	8	16	16	16	16	16	16	16	16
9	9	10	11	12	13	14	15	16	16	16	16	16	16	16	16	16
10	10	10	12	12	14	14	16	16	16	16	26	26	27	27	28	28
11	11	12	12	12	15	16	16	16	16	26	26	26	28	28	30	30
12	12	12	12	12	16	16	16	16	16	26	26	26	28	30	32	32
13	13	14	15	16	16	16	16	16	16	27	28	28	28	32	32	32
14	14	14	16	16	16	16	16	16	16	27	28	30	32	32	32	32
15	15	16	16	16	16	16	16	16	16	28	30	32	32	32	32	32
16	16	16	16	16	16	16	16	16	16	28	30	32	32	32	32	32

2.6 Generalizing Results from Topology

In this section, we review generalizations of the previous results obtained from topology to the arbitrary field case, providing further support for our conjecture that existence of sums-of-squares formulas may be independent of the base field.

In [6], Dugger and Isaksen generalize Hopf's Theorem for general fields of characteristic not 2.

Theorem 2.6.1. *If F is a field of characteristic not equal to 2, and a sums-of-squares formula of type $[r, s, n]$ exists over F, then $\binom{n}{i}$ must be even for $n - r < i < s$.*

The proof follows the same idea as the proof of Hopf's theorem over $\mathbb{R}$, showing that a sums-of-squares formula induces a map on "deleted quadrics" (which take the place of projective space), and considering the motivic cohomology of the deleted quadric. Although the motivic cohomology of the deleted quadric is not known exactly, Dugger and Isaksen were able to use what is known to analyze this map, and thus they arrived at the given condition.

Dugger and Isaksen also proved the following result [7], generalizing the corresponding result using K-theory over $\mathbb{R}$.

Theorem 2.6.2. *Assume F is a field of characteristic not 2. If a sums-of-squares formula of type $[r, s, n]$ exists over F, then $2^{\lfloor \frac{s-1}{2} \rfloor - i + 1}$ divides $\binom{n}{i}$ for $n - i < i \leq \lfloor \frac{s-1}{2} \rfloor$.*

They proved this result using algebraic K-theory in place of topological K-theory.

Note that Dugger and Isaksen don't quite match the result over $\mathbb{R}$. Xie [23] used Hermitian K-theory to improve on their result, and match the result over $\mathbb{R}$:

Theorem 2.6.3. *If a formula of type $[r, s, n]$ exists over a field F of characteristic not 2, then $2^{\phi(r-1)-i+1}$ divides $\binom{n}{i}$ for $n - s < i \leq \phi(r-1)$.*

Dugger and Isaksen also used étale cohomology to generalize another condition on existence of sums-of-squares formulas [8], previously found in characteristic 0 by Davis.

2.7 Motivation

We finish this chapter by listing motivations for further study of sums-of-squares formulas.

For these facts, recall the following definitions:

Definition 2.7.1. A map $f : \mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R}^n$ is *normed* if $\|f(x, y)\| = \|x\|\|y\|$.
If a map $f : \mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R}^n$ is normed and bilinear, then it is nonsingular, bi-skew, and skew-linear. (These weaker conditions enable us to give some of the topological results as equivalent statements.)

f is *nonsingular* if $f(x, y) = 0$ implies $x = 0$ or $y = 0$.

f is *bi-skew* if $f(-x, y) = -f(x, y) = f(x, -y)$.

f is *skew-linear* if it is skew in the first variable and linear in the second.

Now we give motivation for studying sums-of-squares formulas:

- Sums-of-squares formulas are related to the immersion problem: there is an immersion $\mathbb{P}^{r-1} \to \mathbb{R}^{n-1}$ if and only if there is a nonsingular bi-skew map of size $[r, r, n]$. (It is unknown if this equivalent to the existence of a nonsingular bilinear map of size $[r, r, n]$, though a nonsingular bilinear map is automatically bi-skew, so one implication is known)

- Existence of sums-of-squares formulas of type $[n, n, n]$ is equivalent to the existence of an n-dimensional real normed division algebra (as in the example of a $[2, 2, 2]$ formula for $\mathbb{C}$). An early use of sums-of-squares formulas was to show that the only real normed division algebras are $\mathbb{R}$, $\mathbb{C}$, $\mathbb{H}$, and $\mathbb{O}$.

- There is a nonsingular skew-linear map of size $[r, s, n]$ over $\mathbb{R}$ if and only if $n \cdot \xi_{r-1}$ over $\mathbb{P}$ admits s linearly independent sections. Here, $n \cdot \xi_{r-1}$ is the direct sum of n copies of the canonical line bundle on $\mathbb{P}^{r-1}$.

- A normed pairing of size $[r, s, n]$ yields a map

$$H : \mathbb{R}^r \times \mathbb{R}^s \to \mathbb{R} \times \mathbb{R}^n$$

  given by $H(x, y) = (|x|^2 - |y|^2, 2f(x, y))$. This restricts to a Hopf map

$$H : S^{r+s-1} \to S^n$$

  which is a nontrivial quadratic map between spheres. Note that the Hopf map that we get from the formula of type $[2, 2, 2]$ is just the usual Hopf map $S^3 \to S^2$.

- Over arbitrary fields, sums-of-squares formulas are an example of a composition of quadratic forms. Furthermore, it may be the case that existence of a formula of a given type is independent of the base field (for fields of characteristic not 2), in which case we could restrict our study to finite fields, over which formulas could be found with a computer.

CHAPTER 3

Reformulation

In this chapter, which is the basis for the rest of this dissertation, we introduce a reformulation of the question of existence of sums-of-squares as a question in algebraic geometry. This new perspective on sums-of-squares formulas opens up many possible applications of tools from algebraic geometry to the study of sums-of-squares formulas, and the remaining chapters will explore some of these applications.

We reformulate the question of existence of sums-of-squares formulas as one in algebraic geometry, by observing that a sums-of-squares formula is given by the coefficients of $x_j y_k$ in each z_i, which must satisfy certain polynomial equations. Denote by X_{rsn} the scheme defined by these polynomial equations; we will see that X_{rsn} is defined over $\mathbb{Z}$. Then giving a sums-of-squares formula of type $[r, s, n]$ over a ring R is the same as giving an R-point of X_{rsn}.

Denote by A^F_{rsn} the coordinate ring of X_{rsn} over the field F, and let $X^F_{rsn} = \text{Spec } A^F_{rsn}$, which we call the *variety of sums-of-squares formulas of type* $[r, s, n]$ *over* F.

3.1 Reformulation

We would like to consider the question of when sums-of-squares formulas exist, i.e. when, for a fixed field F with $\text{char}(F) \neq 2$, we have

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots + y_s^2) = z_1^2 + \cdots + z_n^2$$

where the z_i are bilinear expressions in the x's and y's with coefficients in F.

We reformulate this question as one about an ideal in a polynomial ring.

We can write

$$z_i = \sum_{j,\,k} \alpha_{ijk} x_j y_k$$

Then, expanding the left side of the formula, we have:

$$\sum_{j,\,k} x_j^2 y_k^2$$

Expanding the right side of the formula, we get:

$$\begin{array}{rcl} \sum_{i,j,k}(\alpha_{ijk}^2 x_j^2 y_k^2) & + & \sum_{i,j<j',k}(2\alpha_{ijk}\alpha_{ij'k} x_j x_{j'} y_k^2) \\ & + & \sum_{i,j,k<k'}(2\alpha_{ijk}\alpha_{ijk'} x_j^2 y_k y_{k'}) \\ & + & \sum_{i,j<j',k<k'}(2(\alpha_{ijk}\alpha_{ij'k'} + \alpha_{ijk'}\alpha_{ij'k}) x_j x_{j'} y_k y_{k'}) \end{array}$$

Grouping terms, we see that the coefficient of $x_j^2 y_k^2$ on the right side is:

$$\sum_i \alpha_{ijk}^2$$

the coefficient of $x_j x_{j'} y_k^2$ where $j < j'$ is:

$$\sum_i 2\alpha_{ijk}\alpha_{ij'k}$$

the coefficient of $x_j^2 y_k y_{k'}$ where $k < k'$ is:

$$\sum_i 2\alpha_{ijk}\alpha_{ijk'}$$

and the coefficient of $x_j x_{j'} y_k y_{k'}$ where $j < j'$ and $k < k'$ is:

$$\sum_i 2(\alpha_{ijk}\alpha_{ij'k'} + \alpha_{ijk'}\alpha_{ij'k})$$

Comparing these expressions with the coefficients on the left side of the formula, we see that existence of a sums-of-squares formula over F is equivalent to the existence of $\alpha_{ijk} \in F$ satisfying

$$\left\{ \begin{array}{ll} \sum_i \alpha_{ijk}^2 = 1 & \text{for all } j, k \\ \sum_i \alpha_{ijk}\alpha_{ij'k} = 0 & \text{for } j < j' \text{ and all } k \\ \sum_i \alpha_{ijk}\alpha_{ijk'} = 0 & \text{for all } j \text{ and } k < k' \\ \sum_i (\alpha_{ijk}\alpha_{ij'k'} + \alpha_{ijk'}\alpha_{ij'k}) = 0 & \text{for } j < j' \text{ and } k < k' \end{array} \right\}$$

Note that we can drop the coefficient 2, since $\mathrm{char}(F) \neq 2$.

Changing notation, we are asking if the set of polynomials

$$\left\{ \begin{array}{ll} \sum_i (x_{ijk}^2) - 1 & \text{for all } j,k \\ \sum_i x_{ijk}x_{ij'k} & \text{for } j < j' \text{ and all } k \\ \sum_i x_{ijk}x_{ijk'} & \text{for all } j \text{ and } k < k' \\ \sum_i (x_{ijk}x_{ij'k'} + x_{ijk'}x_{ij'k}) & \text{for } j < j' \text{ and } k < k' \end{array} \right\}$$

has a zero.

Let X_{rsn} denote the scheme defined by this set of polynomials, and let I be the ideal generated by this set of polynomials in $F[\{x_{ijk}\}]$.

The set of polynomials has a zero in an algebraic closure of F if and only if I is a proper ideal. Thus, we have the following reformulation for the existence of sums-of-squares formulas:

Proposition 3.1.1. *Let I be the ideal generated by the following set of polynomials in $S = F[\{x_{ijk}\}]$ (where $1 \leq i \leq n$, $1 \leq j \leq r$, $1 \leq k \leq s$):*

$$\left\{ \begin{array}{ll} \sum_i (x_{ijk}^2) - 1 & \textit{for all } j,k \\ \sum_i x_{ijk}x_{ij'k} & \textit{for } j < j' \textit{ and all } k \\ \sum_i x_{ijk}x_{ijk'} & \textit{for all } j \textit{ and } k < k' \\ \sum_i (x_{ijk}x_{ij'k'} + x_{ijk'}x_{ij'k}) & \textit{for } j < j' \textit{ and } k < k' \end{array} \right\}$$

A sums-of-squares formula of type $[r,s,n]$ exists over an algebraic closure of F if and only if $I \subset S$ is a proper ideal.

We set the notation $A_{rsn}^F = S/I$, where S and I are as above, and write $X_{rsn}^F = \mathrm{Spec}\ (A_{rsn}^F)$.

In particular, in the algebraically closed case, this reformulation can be stated as follows:

Proposition 3.1.2. *Let F be an algebraically closed field. Then a sums-of-squares formula of type $[r,s,n]$ exists over F if and only if $A_{rsn}^F \neq 0$.*

By the Hilbert Nullstellensatz, for F an algebraically closed field, this is also equivalent to X_{rsn}^F begin a nonempty scheme, and to $X_{rsn}(F)$ being nonempty.

3.2 Consequences

We have reformulated the question of existence of sums-of-squares formulas as a question in algebraic geometry, and this opens up a myriad of possible applications of tools of algebraic geometry to the subject. It also raises many new questions about this variety, including:

- Is it empty? For what F does it have F-rational points? (i.e., existence of sums-of-squares formulas)
- Is it flat?
- Is it irreducible? Connected?
- Is it smooth? Non-singular? Regular?
- What is its dimension?
- If it has multiple components, how do they compare? Are there isolated points? How do sums-of-squares formulas from different components differ?

Unfortunately, given the huge number of variables and polynomials necessary to define the scheme, these are very difficult questions to answer.

In the remainder of this thesis, we apply methods of algebraic geometry to extract information about this scheme and thus produce results about sums-of-squares formulas.

In chapters 4 and 5, we focus on how the existence of a sums-of-squares formula depends on the base field. In chapter 4, we prove that, for algebraically closed fields, a formula exists over a field of characteristic 0 if and only if one exists over a field of characteristic p, for all but finitely many p. Furthermore, a sums-of-squares formula exists over some finite field, and we give an upper bound for the degree of this field. These results are obtained using number theory, in particular, an argument considering the coefficients of the minimal polynomial of the coefficients of the sums-of-squares formula, and later the zeta function. In chapter 5, we show that for "large enough" p, for algebraically closed fields, the characteristic 0 and

characteristic p cases are equivalent. Thus, we obtain an explicit bound on the finitely many p from Chapter 4. We give an extremely large bound for "large enough" by analyzing the coefficients that can appear in a Gröbner basis for the ideal defining our scheme. We thus establish that the existence of sums-of-squares formulas over algebraically closed fields is computable.

In chapter 6, we begin to explore the shape of the variety by defining an action of orthogonal groups, which raises additional questions about the structure of this action. We use this group action to establish a lower bound for the dimension when the variety is non-empty.

In chapter 7, we consider computational methods for finding sums-of-squares formulas over finite fields and over the integers. These computational methods take advantage of the group action defined in chapter 6, and the results from chapter 4 increase their significance, since we have shown that sums-of-squares formulas can (theoretically) always be detected over finite fields, although the bounds are too large to currently be feasible.

CHAPTER 4

Partial Independence from the Base Field

In this chapter, we show that for algebraically closed fields, existence in characteristic 0 and existence in characteristic p are equivalent for all but finitely many p, as well as some related results. This uses the reformulation introduced in the previous chapter, which characterizes existence of sums-of-squares formulas as a question in algebraic geometry.

As a first step towards the main theorem we observe that if $F \subset K$ are algebraically closed fields, there is a sums-of-squares formula of type $[r, s, n]$ over F if and only if there is one over K. This is a consequence of the Hilbert Nullstellensatz.

Using a reduction argument, we also show that if there is no sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{Q}}$, then there is no sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{F}}_p$ for all but finitely p. This gives us a half of our main theorem.

To finish our main theorem, taking α_{ijk} to give a sums-of-squares formula over $\bar{\mathbb{Q}}$, we consider the coefficients of the minimal polynomials of the α_{ijk} to show that if there is a sums-of-squares formula of type $[r, s, n]$ exists over $\bar{\mathbb{Q}}$, then a sums-of-squares formula of the same type exists over $\bar{\mathbb{F}}_p$ for all but finitely many primes p.

Thus, we have that, for algebraically closed fields, existence of a sums-of-squares formula over fields of characteristic 0 and characteristic p are equivalent for all but finitely many p.

As a corollary to this theorem, we show that if a sums-of-squares formula exists over any field, then a sums-of-squares formula exists over some finite field $\mathbb{F}_{p^m}$. This follows from the main theorem by noting that a formula consists of only finitely many coefficients. In the second section, we improve on this result by providing an upper bound for how large such a finite field must be. This is accomplished by considering the zeta function and applying a

theorem due to Bombieri [3].

4.1 Results

We begin with an immediate consequence of our reformulation and the Hilbert Nullstellensatz, which allows us to consider only $\bar{\mathbb{Q}}$ and $\bar{\mathbb{F}}_p$ for p prime.

Theorem 4.1.1. *If $F \subset K$ are algebraically closed fields, there is a sums-of-squares formula of type $[r, s, n]$ over F if and only if there is one over K.*

In particular, in determining existence of formulas over algebraically closed fields, it suffices to consider the fields $\bar{\mathbb{Q}}$ and $\bar{\mathbb{F}}_p$ for p prime.

Consequently,

Corollary 4.1.2. *A sums of squares formula exists over $\mathbb{C}$ if and only if it exists over $\bar{\mathbb{Q}}$.*

To prove half of our main theorem, we make a reduction argument about what happens if there are no formulas of type $[r, s, n]$ over $\bar{\mathbb{Q}}$ (or, equivalently, any algebraically closed field of characteristic 0):

Proposition 4.1.3. *If there is no sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{Q}}$, then there is no sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{F}}_p$ for all but finitely many p.*

Proof. Suppose there is no sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{Q}}$. Then

$$A^{\mathbb{Q}}_{rsn} = A^{\mathbb{Z}}_{rsn} \otimes_{\mathbb{Z}} \mathbb{Q} = 0$$

so for all $f \in A^{\mathbb{Z}}_{rsn}$, we must have $af = 0$ for some $0 \neq a \in \mathbb{Z}$. In particular, there is $a \in \mathbb{Z}$ such that $a \cdot 1 = 0$. Then $af = 0$ for all f, so we can find one a that works for all f.

Now, let p be a prime not dividing a. Then a is invertible in $\mathbb{Z}/p\mathbb{Z}$, so

$$A^{\mathbb{Z}}_{rsn} \otimes_{\mathbb{Z}} \mathbb{Z}/p\mathbb{Z} = 0$$

Thus, for all primes p not dividing a, there is no sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{F}}_p$.

There are only finitely many primes dividing a, so this completes the proof. □

Now, we will prove the second part of our main theorem:

Proposition 4.1.4. *If there is a sums-of-squares formula of type $[r, s, n]$ over any field of characteristic 0, then there is a sums-of-squares formula of type $[r, s, n]$ over every algebraically closed field of characteristic p for all but finitely many p.*

Note that if a sums-of-squares formula exists over a field F of characteristic 0, then that same formula is valid over the algebraic closure $\bar{F}$. Then, by our previous theorem, there is a sums-of-squares formula over $\bar{\mathbb{Q}}$.

Proof. Suppose there is a sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{Q}}$ with coefficients α_{ijk}. Let K be the number field generated by the α_{ijk}, and O_K the ring of algebraic integers in K. Since there are finitely many α_{ijk}, there are finitely many coefficients in their minimal polynomials. Let S be the set of primes that are divisors of the denominators of the coefficients of these minimal polynomials. Then, for prime p not in S, for every prime ideal P in O_K lying over (p), the coefficients α_{ijk} are integral at P. Reducing modulo P, we obtain a sums-of-squares formula over the field O_K/P, which is a finite field extension of $\mathbb{F}_p$. Hence there is a sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{F}}_p$ for p not in the finite set S. □

Putting these two propositions together, we have shown:

Theorem 4.1.5. *For algebraically closed fields, existence of sums-of-squares formulas of type $[r, s, n]$ over fields of characteristic 0 and p are equivalent for all but finitely many p.*

Of course, we would like to be able to eliminate the condition "for all but finitely many p."

Considering the scheme $X^{\mathbb{Z}}_{rsn}$ over $\operatorname{Spec} \mathbb{Z}$, we have shown that the generic fiber is nonempty if and only if "almost all" fibers are nonempty. If we could somehow show that

this scheme is flat, then since it would have open image, if the fiber over a prime p were nonempty, then all fibers would be nonempty. Showing in addition that this image is closed would establish the independence of existence of a sums-of-squares formula from the base field in the algebraically closed case. However, showing that this scheme is flat seems to be very difficult.

We also have the following corollary:

Corollary 4.1.6. *If a sums-of-squares formula of type $[r, s, n]$ exists over a field F, then a sums-of-squares formula of type $[r, s, n]$ exists over some finite field.*

Proof. If F has characteristic 0, then there is a sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{F}}_p$ for some p.

If F has characteristic q, then there is a sums-of-squares formula of type $[r, s, n]$ over $\bar{F}$, hence over $\bar{\mathbb{F}}_q$.

In either case, we have a sums-of-squares formula of type $[r, s, n]$ over some $\bar{\mathbb{F}}_q$. This sums-of-squares formula is given by finitely many coefficients α_{ijk}, so we have a sums-of-squares formula over $\mathbb{F}_q[\{\alpha_{ijk}\}]$. The α_{ijk} are algebraic over $\mathbb{F}_q$, so this is a finite extension of a finite field. Thus we have a sums-of-squares formula of type $[r, s, n]$ over a finite field. □

This corollary means that that sums-of-squares formulas can, in principle, be detected by checking over finite fields, and this can be done with a computer search. We do not know a priori how large of a field would be needed. The next section is dedicated to establishing a bound on the degree of this field. However, this bound is too large to be computationally feasible for conducting computer searches for formulas. In Chapter 6, we obtain an (also unwieldy) bound on how large the "finitely many" p can be. Computer searches for formulas will be discussed in Chapter 7.

4.2 Degree Bound

In this section, we show that for fixed $[r, s, n]$ and prime p, there is d such that if there is no sums-of-squares formula of type $[r, s, n]$ over $\mathbb{F}_{p^k}$ for $k < d$, then there is no sums-of-squares formula over $\bar{\mathbb{F}}_p$. Thus, we can theoretically check computationally if a sums-of-squares formula exists over $\bar{\mathbb{F}}$, since there are only finitely many possible formulas over $\mathbb{F}_{p^k}$. However, the value of d is too large for this to be computationally feasible.

This result follows immediately from a theorem bounding the total degree of the zeta function due to Bombieri [3]. We review the relevant material on zeta functions before giving our result. This material can be found in [21].

Let $\mathbb{F}_p$ be the field with p elements, for p prime. Let X be an algebraic set over $\mathbb{F}_p$, we assume that X is affine and defined by m polynomials in n variables $f_1, ..., f_m \in \mathbb{F}_p[x_1, ..., x_n]$, so

$$X(\mathbb{F}_p) = \{x \in \mathbb{F}_p^n \mid f_1(x) = \cdots = f_m(x) = 0\}$$

Let d be the maximum total degree among the polynomials f_i.

Fix an algebraic closure $\bar{\mathbb{F}}_p$, and let $\mathbb{F}_{p^k}$ denote the unique subfield of $\bar{\mathbb{F}}_p$ with p^k elements. $\#X(\mathbb{F}_{p^k})$ denotes the number of $\mathbb{F}_{p^k}$-rational points on X.

Definition 4.2.1. The *zeta function* of X is the generating function

$$Z(X) = Z(X, T) = exp\left(\sum_{k=1}^{\infty} \frac{T^k}{k} \#X(\mathbb{F}_{p^k})\right) \in 1 + T\mathbb{Z}[T]$$

As conjectured by Weil [22] and proved by Dwork [9] and Grothendieck [11], this is a rational function:

Theorem 4.2.2. *$Z(X)$ is a rational function, so it can be written*

$$Z(X, T) = \frac{R_1(X, T)}{R_2(X, T)}$$

with $R_1, R_2 \in \mathbb{Z}[T]$.

In fact, we can take $R_1, R_2 \in 1 + T\mathbb{Z}[T]$.

The following theorem due to Bombieri [3] gives an upper bound on the total degree $\deg R_1 + \deg R_2$ of the zeta function.

Theorem 4.2.3. *With R_1, R_2, d, n, m as above, we have:*

$$deg\ R_1 + deg\ R_2 < (4d+9)^{n+m}$$

The following algorithm, which can be found in [21] shows that we can compute the zeta function by computing only finitely many of the $\#X(\mathbb{F}_{p^k})$. This determines the rest of the $\#X(\mathbb{F}_{p^k})$.

Let D_1 and D_2 be upper bounds for the degrees of R_1 and R_2, where $Z(X,T) = \frac{R_1(X,T)}{R_2(X,T)}$. For example, take $D_1 = D_2 = (4d+9)^{n+m}$ as given by the previous theorem.

Then, by counting $X(\mathbb{F}_{p^k})$ for $k \leq D_1 + D_2$ (there are only finitely many elements in $\mathbb{F}_{p^k}^n$), we can compute the first $D_1 + D_2 + 1$ terms in the power series $Z(X,T)$:

$$Z(X,T) = 1 + z_1 T + z_2 T^2 + \cdots + z_{D_1+D_2} T^{D_1+D_2} + \cdots$$

Write

$$R_1(X) = 1 + a_1 T + \cdots + a_{D_1} T^{D_1}$$
$$R_2(X) = 1 + b_1 T + \cdots + b_{D_2} T^{D_2}$$

Our goal is to determine the coefficients a_i and b_i.

We have

$$R_2(X)Z(X) \equiv R_1(X) \pmod{T^{D_1+D_2+1}}$$

This gives a linear system of equations in the a_i and b_i. We know that it has a solution, which can be found using linear algebra. Thus the zeta function is determined.

This means that once we know $\#X(\mathbb{F}_{p^k})$ for $k \leq D_1 + D_2$, the zeta function is determined, so $\#X(\mathbb{F}_{p^k})$ for $k > D_1 + D_2$ are determined. In particular, if $\#X(\mathbb{F}_{p^k}) = 0$ for $k \leq D_1 + D_2$, then $\#X(\mathbb{F}_{p^k}) = 0$ for all k, and $\#X(\bar{\mathbb{F}}_p) = 0$. Applying this to the case of sums-of-squares formulas, we have the following result:

Theorem 4.2.4. *Fix $[r, s, n]$ and prime p. If there is a sums-of-squares formula of type $[r, s, n]$ over $\bar{\mathbb{F}}_p$, then there is a sums-of-squares formula of type $[r, s, n]$ over $\mathbb{F}_{p^k}$ for some $k \leq 2 \cdot 17^{rsn+r^2s^2}$.*

Thus, in principle, sums-of-squares formulas can be detected using searches over finite fields. However, the size of the fields that would be required make this unrealistic.

4.3 Discussion

We have shown that, fixing $[r, s, n]$ and considering algebraically closed fields, existence of a formula in characteristic 0 and p are equivalent for all but finitely many p. We would like to be able to remove the "for all but finitely many p" condition and establish independence from the base field in the algebraically closed case. This may be possible by showing that the scheme $X^{\mathbb{Z}}_{rsn}$ is flat over $\mathbb{Z}$ and has closed image. Unfortunately, this remains beyond reach. However, we are able to prove a weaker result in the next chapter, establishing a partial converse.

We have shown that when we fix $[r, s, n]$, for "large enough" p, existence of a sums-of-squares formula of type $[r, s, n]$ over an algebraically closed field is independent of the characteristic. In the next Chapter, we use Gröbner bases to obtain an (atrocious) upper bound for this value of p.

We also showed that if a sums-of-squares formula of type $[r, s, n]$ exists over some field, then a formula of the same type exists over a finite field. We derived a bound for the degree of this field. Thus we showed that existence of sums-of-squares formulas can always be detected over finite fields, and we have a bound on how large these finite fields can be. Then, theoretically, we can computationally check existence of sums-of-squares formulas by checking finitely many possibilities. Unfortunately, the bounds we have are too large for this to be practical. In Chapter 7 we discuss computer searches for formulas over finite fields and over the integers, which can provide new examples of sums-of-squares formulas, even though we cannot efficiently use these methods to determine existence or non-existence of formulas

over arbitrary fields.

CHAPTER 5

A Lower Bound on Independence

In the previous chapter, we proved that, for algebraically closed field, existence of a sums-of-squares formula over fields of characteristic 0 and characteristic p are equivalent, for all but finite many p. In this chapter, we show equivalence for "large enough p," providing an explicit bound.

Recall that we have found the following characterization of the existence of sums-of-squares formulas:

Proposition 5.0.1. *Let I be the ideal generated by the following set of polynomials in $S = F[\{x_{ijk}\}]$ (where $1 \leq i \leq n$, $1 \leq j \leq r$, $1 \leq k \leq s$):*

$$\left\{ \begin{array}{ll} \sum_i (x_{ijk}^2) - 1 & \text{for all } j, k \\ \sum_i x_{ijk} x_{ij'k} & \text{for } j < j' \text{ and all } k \\ \sum_i x_{ijk} x_{ijk'} & \text{for all } j \text{ and } k < k' \\ \sum_i (x_{ijk} x_{ij'k'} + x_{ijk'} x_{ij'k}) & \text{for } j < j' \text{ and } k < k' \end{array} \right\}$$

A sums-of-squares formula of type $[r, s, n]$ exists over an algebraic closure of F if and only if $I \subset S$ is a proper ideal.

We have thus reduced the question of existence of sums-of-squares formulas to the question of whether or not a certain ideal is proper. This can be decided using Gröbner bases, although not efficiently enough to be useful in this case.

However, Gröbner bases can be computing using Buchberger's algorithm, and by analyzing the coefficients which can appear at each step in Buchberger's algorithm, we are able to prove:

Theorem 5.0.2. *Fix r, s, n. For "large enough" p, a sums-of-squares formula of type $[r, s, n]$ exists over an algebraically closed field of characteristic 0 if and only if a sums-of-squares formula of type $[r, s, n]$ over an algebraically closed field of characteristic p.*

By analyzing coefficients, we are able to give an explicit bound for p for which this is true.

We begin by reviewing the relevant results on Gröbner bases, which can be found in Eisenbud's Commutative Algebra book [10] and Dubé's paper [5] analyzing the degrees of elements in a Gröbner basis.

5.1 Gröbner Bases and Buchberger's Algorithm

Let $S = k[x_1, ..., x_r]$ be a polynomial ring over a field k. A *monomial order* on S is a total order $>$ on the monomials of S such that if m_1, m_2 are monomials of S and $n \neq 1$ is a monomial of S, then

$$m_1 > m_2 \text{ implies } nm_1 > nm_2 > m_2.$$

For the situation of sums-of-squares formulas, we find it most convenient to work with degree reverse lexicographic order. By deg, we mean the total degree of the monomial.

Definition 5.1.1. (Degree Reverse Lexicographic Order) Let $m = x_1^{a_1} \cdots x_r^{a_r}$ and $n = x_1^{b_1} \cdots x_r^{b_r}$ be monomials in S.

$m >_{\text{dp}} n$ if

$$\deg m > \deg n$$

or if

$$\deg m = \deg n \text{ and } a_i < b_i \text{ for the last index } i \text{ with } a_i \neq b_i.$$

Since we will only work with degree reverse lexicographic order, we will simply write $>$ for $>_{\text{dp}}$.

If $>$ is a monomial order, for any $f \in S$ we define the *initial term* of f, which we denote $\text{in}(f)$, to be the greatest term of f with respect to the order $>$. If $I \subset S$ is an ideal, we define $\text{in}(I)$ to be the monomial ideal generated by the initial terms of elements in I.

We are now in a position to define a Gröbner basis for an ideal $I \subset S$.

Definition 5.1.2. Let $I \subset S = k[x_1, ..., x_r]$ be an ideal. A Gröbner basis for I with respect to an order $>$ is a set of elements $g_1, ..., g_t \in I$ such that

1. $g_1, ..., g_t$ generate I.
2. $\text{in}(g_1), ..., \text{in}(g_t)$ generate $\text{in}(I)$.

One advantage of Gröbner bases is that they enable us to easily decide if two ideals are equal, a result of the following lemma.

Lemma 5.1.3. *If $I \subset J \subset S$ are ideals and $in(I) = in(J)$ with respect to a monomial order, then $I = J$.*

Thus, once we have found Gröbner bases for I and J, we can decide if they are equal simply by comparing the initial terms of the Gröbner bases, which are monomials.

Gröbner bases can be computed using Buchberger's Algorithm, for which we first require the Division Algorithm.

Proposition 5.1.4. *Let $S = k[x_1, ..., x_r]$ be a polynomial ring over a field k with a monomial order $>$. If $f, g_1, ..., g_t \in S$, then there is an expression*

$$f = \sum f_i g_i + f' \text{ with } f', f_i \in S,$$

where none of the monomials of f' is in $(in(g_1), ..., in(g_t))$ and

$$in(f) \geq in(f_i g_i)$$

for every i. f' is called a remainder *of f with respect to $g_1, ..., g_t$. The expression $f = \sum f_i g_i + f'$ as above is called a* standard expression *for f in terms of the g_i.*

Note that, unlike the single variable case, the standard expression and remainder are not necessarily unique.

The Division Algorithm gives us an explicit construction of a standard expression.

Proposition 5.1.5. ***(Division Algorithm)*** *Let $S = k[x_1, ..., x_r]$ be a polynomial ring over a field k with a monomial order.If $f, g_1, ..., g_t \in S$, then we can produce a standard expression*

$$f = \sum m_u g_{s_u} + f'$$

for f with respect $g_1, ..., g_t$. We do this by defining the indices s_u and terms m_u inductively. Write

$$f'_p = f - \sum_{u=1}^{p} m_u g_{s_u} \neq 0$$

and m is the maximal term of f'_p that is divisible by some $in(g_i)$, then we take

$$s_{p+1} = i$$

$$m_{p+1} = m/in(g_i)$$

The process ends when $f'_p = 0$ or when no $in(g_i)$ divides a monomial of f'_p. The remainder f' is the last f'_p produced.

This algorithm ends after finitely many steps since the maximal term of f'_p divisible by some $\text{in}(g_i)$ decreases at each step.

Buchberger's Criterion allows us to determine if a set of elements is a Gröbner basis.

Proposition 5.1.6. ***(Buchberger's Criterion)*** *Let $S = k[x_1, ..., x_r]$ be a polynomial ring with k a field, and $g_1, ..., g_t \in S$ nonzero elements. For each pair of indices i, j, define*

$$m_{ij} = in(g_i)/GCD(in(g_i), in(g_j)) \in S$$

Then choose a standard expression

$$m_{ji}g_i - m_{ij}g_j = \sum f_u^{(ij)} g_u + h_{ij}$$

for $m_{ji}g_i - m_{ij}g_j$ *with respect to* $g_1, ..., g_t$.

Then the elements $g_1, ..., g_t$ *form a Gröbner basis if and only if* $h_{ij} = 0$ *for all* i *and* j.

This criterion leads to Buchberger's Algorithm, which allows us to construct Gröbner bases.

Proposition 5.1.7. ***(Buchberger's Algorithm)*** *In the situation of Buchberger's Criterion, suppose* $I \subset S$ *is an ideal, and let* $g_1, ..., g_t \in I$ *be a set of generators of* I. *If all the* $h_{ij} = 0$, *then the* g_i *form a Gröbner basis for* I. *If some* $h_{ij} \neq 0$, *then replace* $g_1, ..., g_t$ *with* $g_1, ..., g_t, h_{ij}$, *and repeat the process. The ideal generated by the initial terms of* $g_1, ..., g_t, h_{ij}$ *is strictly larger than the ideal generated by the initial terms of* $g_1, ..., g_t$, *so this process terminates after finitely many steps (since* S *is noetherian).*

Computing a Gröbner basis would give us a clearer idea of the structure of the scheme of sums-of-squares formula. However, this is only computationally feasible for very small values of r, s, n. The main question we would like to answer is when 1 is in the relevant ideal (since when the ideal is not proper, no sums-of-squares formula exists). By studying the coefficients that can arise in the Division Algorithm and Buchberger's Algorithm, we show that for "large enough" p, the algebraically closed cases in characteristic 0 and characteristic p coincide. This is carried out in the next section.

In order to establish this upper bound, we need some understanding of the complexity of Buchberger's Algorithm. This is provided by [5], in which it is shown:

Theorem 5.1.8. *Let* B *be a set of polynomials with Gröbner basis* G. *If there are* n *variables and the polynomials in* B *have total degree* $\leq d$, *then the degree of the polynomials in* G *is bounded by*

$$2\left(\frac{1}{2}d^2 + d\right)^{2^{n-1}}$$

In the case of sums-of-squares formulas of type $[r, s, n]$, we are considering a set B of polynomials in rsn variables with total degree ≤ 2. Thus the degree of the polynomials in a Gröbner basis is bounded by $2 \cdot 4^{2^{rsn-1}}$.

5.2 Analyzing Coefficients in the Division Algorithm

We have that a sums-of-squares formula of type $[r, s, n]$ exists over an algebraically closed field F if and only if the ideal I generated by

$$\left\{ \begin{array}{ll} \sum_i (x_{ijk}^2) - 1 & \text{for all } j, k \\ \sum_i x_{ijk} x_{ij'k} & \text{for } j < j' \text{ and all } k \\ \sum_i x_{ijk} x_{ijk'} & \text{for all } j \text{ and } k < k' \\ \sum_i (x_{ijk} x_{ij'k'} + x_{ijk'} x_{ij'k}) & \text{for } j < j' \text{ and } k < k' \end{array} \right\}$$

is proper in $S = F[\{x_{ijk}\}]$. This can be determined by computing a Gröbner basis. Note that the polynomials in this generating set all have coefficients ± 1 or 0. We will carefully track the coefficients of polynomials over $\mathbb{Q}$ through the division algorithm and Buchberger's algorithm in order to obtain an upper bound on p for which there can be a difference between the characteristic 0 and characteristic p cases.

For $\frac{a}{b} \in \mathbb{Q}$ with $a, b \in \mathbb{Z}$ relatively prime, we set the notation

$$P\left(\frac{a}{b}\right) = \max\{|a|, |b|\}$$

We prove some basic properties of P before proceeding with the analysis of Buchberger's algorithm.

Proposition 5.2.1. *Let $x, y \in \mathbb{Q}$.*

1. $P\left(\frac{1}{x}\right) = P(x)$
2. $P(-x) = P(x)$
3. $P(xy) \leq P(x)P(y)$

4. $P(x+y) \leq 2P(x)P(y)$

Proof. (1) and (2) are obvious.

Write $x = \frac{a}{b}$ with $a, b \in \mathbb{Z}$ relatively prime and $y = \frac{c}{d}$ with $c, d \in \mathbb{Z}$ relatively prime.

$$\begin{aligned} P(xy) &= P\left(\frac{ac}{bd}\right) \\ &= \max\{|ac|, |bd|\} \\ &\leq \max\{|a|, |b|\} \cdot \max\{|c|, |d|\} \\ &= P(x)P(y) \\ P(x+y) &= P\left(\frac{ad+bc}{bd}\right) \\ &= \max\{|ad+bc|, |bd|\} \\ &\leq 2\max\{|a|, |b|\} \cdot \max\{|c|, |d|\} \\ &= 2P(x)P(y) \end{aligned}$$

□

For simplicity, we will say that a polynomial f has coefficients with P bounded by M if for every coefficient a_i of f, we have $P(a_i) \leq M$.

Proposition 5.2.2. *Suppose we have $g_1, ..., g_t \in S = \mathbb{Q}[x_1, ..., x_r]$. Suppose further that every g_i has total degree $\leq t$ and has coefficients with P bounded by some number M. Then the h_{ij} as computed in Buchberger's criterion have coefficients with P bounded by*

$$2^{3 \cdot 2^p - 2} M^{5 \cdot 2^p - 3}$$

where $p = \begin{pmatrix} rsn + 2t \\ 2t \end{pmatrix}$.

Proof. We have

$$m_{ij} = \text{in}(g_i)/GCD(\text{in}(g_i), \text{in}(g_j))$$

For convenience, we take the GCD to have coefficient 1. Then, since $\text{in}(g_i)$ has coefficient with P bounded by M, m_{ij} has the same coefficient with P bounded by M.

Then $m_{ji}g_i$ has coefficients with P bounded by M^2, so $m_{ji}g_i - m_{ij}g_j$ has coefficients with P bounded by $2M^2$. Note that the total degree of $m_{ji}g_i - m_{ij}g_j$ is $\leq 2t$.

We would like a bound on h_{ij} in the standard expression

$$m_{ji}g_i - m_{ij}g_j = \sum f_u^{(ij)} g_u + h_{ij}$$

which we achieve in the following lemma, and the proposition follows immediately. □

Lemma 5.2.3. *Suppose we have $f, g_1, ..., g_t \in S$, where f has coefficients with P bounded by $2M^2$ and every g_i has coefficients with P bounded by M. Suppose also that f has total degree d. Then, in any standard expression*

$$f = \sum m_u g_{s_u} + f'$$

the f' has coefficients with P bounded by

$$2^{3\cdot 2^p - 2} M^{5\cdot 2^p - 3}$$

where $p = \begin{pmatrix} rsn + d \\ d \end{pmatrix}$.

Proof. For the first step of the division algorithm, we take m to be the maximal term of f that is divisible by some $\mathrm{in}(g_i)$ and choose

$$s_1 = i$$
$$m_1 = m/\mathrm{in}(g_i)$$

m has coefficient with P bounded by $2M^2$ and $\mathrm{in}(g_i)$ has coefficient with P bounded by M, so m_1 has coefficient with P bounded by $2M^3$.

Then

$$f_1' = f - m_1 g_{s_1}$$

has coefficients with P bounded by $2 \cdot 2M^2 \cdot 2M^3 M = 8M^6$.

Suppose f'_p has coefficients with P bounded by N. Let m be the maximal term of f that is divisible by some $\text{in}(g_i)$, and choose

$$s_{p+1} = i$$

$$m_{p+1} = m/\text{in}(g_i)$$

m has coefficient with P bounded by N and $\text{in}(g_i)$ has coefficient with P bounded by M, so m_{p+1} has coefficient with P bounded by $2M^2N$.

Then

$$f'_{p+1} = f'_p - m_{p+1}g_{s_{p+1}}$$

has coefficients with P bounded by $2 \cdot N \cdot 2M^2NM = 4M^3N^2$.

Observe that the recursion $a_{n+1} = ca_n^2$ has closed form given by $a_n = c^{2^n-1}a_0^{2^n}$.

Thus, since $f'_0 = f$ has coefficients with P bounded by $2M^2$, we have that f'_p has coefficients with P bounded by $(4M^3)^{2^p-1}(2M^2)^{2^p} = 2^{3\cdot 2^p-2}M^{5\cdot 2^p-3}$.

We now consider how many steps must be taken in the division algorithm. Now, $\text{in}(f'_p) < \text{in}(f'_{p-1})$ (in the monomial order) and f has total degree t, so the total number of steps to produce the standard expression must be at most the number of monomials of total degree $\leq d$. (Note that this depends on us having chosen a homogeneous order.) In rsn variables, this is given by:

$$p = \begin{pmatrix} rsn + d \\ d \end{pmatrix}$$

□

We now have a bound on P in the h_{ij} computed in Buchberger's Criterion. The next step is to consider how this develops as we proceed through Buchberger's Algorithm in the case of sums-of-squares formulas.

Proposition 5.2.4. *Suppose $g_1, ..., g_t$ generate an ideal I. Suppose also that the g_i all have total degree 2 and have coefficients with P bounded by 1. After m steps of Buchberger's*

Algorithm, the candidates for a Gröbner basis all have coefficients with P bounded by

$$\left(2^{3\cdot 2^p-2}\right)^{\left((5\cdot 2^p-3)^{m+1}-1\right)/(5\cdot 2^p-4)}$$

where $p = \begin{pmatrix} rsn + 4^{2^{n-1}+1} \\ 4^{2^{n-1}+1} \end{pmatrix}$.

Proof. From Dubé's result [5], we have that the degree of the polynomials in a Gröbner basis for the ideal I is bounded by $2 \cdot 4^{2^{n-1}}$.

After m steps of Buchberger's Algorithm, suppose the candidates for a Gröbner basis all have coefficients with P bounded by a_m. The previous proposition gives us the following recursion:

$$a_m = 2^{3\cdot 2^p-2} \left(a_{m-1}\right)^{5\cdot 2^p-3}$$

where $p = \begin{pmatrix} rsn + 2\left(2 \cdot 4^{2^{n-1}}\right) \\ 2\left(2 \cdot 4^{2^{n-1}}\right) \end{pmatrix} = \begin{pmatrix} rsn + 4^{2^{n-1}+1} \\ 4^{2^{n-1}+1} \end{pmatrix}$

A closed form of this recursion with $a_0 = 1$ is given by:

$$a_m = \left(2^{3\cdot 2^p-2}\right)^{\left((5\cdot 2^p-3)^{m+1}-1\right)/(5\cdot 2^p-4)}$$

□

Finally, by establishing an upper bound on the number of steps in Buchberger's Algorithm in our case, we arrive at our result.

Theorem 5.2.5. *Fix r, s, n. Consider a prime p with*

$$p > \left(2^{3\cdot 2^q-2}\right)^{\left((5\cdot 2^q-3)^{m+1}-1\right)/(5\cdot 2^q-4)}$$

where $q = \begin{pmatrix} rsn + 4^{2^{n-1}+1} \\ 4^{2^{n-1}+1} \end{pmatrix}$ *and* $m = \begin{pmatrix} rsn + 2 \cdot 4^{2^{n-1}} \\ 2 \cdot 4^{2^{n-1}} \end{pmatrix}$.

Then a sums-of-squares formula of type $[r, s, n]$ exists over an algebraically closed field of characteristic 0 if and only if a sums-of-squares formula of type $[r, s, n]$ exists over an algebraically closed field of characteristic p.

Proof. Once again, we use that the degree of polynomials in a Gröbner basis is bounded by $2 \cdot 4^{2^{n-1}}$. Note that, in each step Buchberger's Algorithm, the ideal generated by the initial terms of $g_1, ..., g_t, h_{ij}$ is strictly larger than the ideal generated by the initial terms $g_1, ..., g_t$. Since this is a monomial ideal, this means that the number of steps in Buchberger's algorithm is bounded by the number of monomials of degree $\leq 2 \cdot 4^{2^{n-1}}$.

Thus the number of steps in Buchberger's Algorithm is bounded by:

$$\begin{pmatrix} rsn + 2 \cdot 4^{2^{n-1}} \\ 2 \cdot 4^{2^{n-1}} \end{pmatrix}$$

Consider I as above, the relevant ideal for sums-of-squares formulas. Every polynomial in a Gröbner basis for I has coefficients with P bounded by 5.2.5. Then, for

$$p > \left(2^{3 \cdot 2^q - 2}\right)^{\left((5 \cdot 2^q - 3)^{m+1} - 1\right)/(5 \cdot 2^q - 4)}$$

(as in the statement of the theorem), we have that every coefficient of polynomials in the Gröbner basis is nontrivial if and only if it is nontrivial when reduced modulo p. This is because the coefficients are all rational numbers, and neither the numerator nor denominator will become 0 modulo p.

Recall that $I = S = F[\{x_{ijk}\}]$ if and only if no sums-of-squares formula of type $[r, s, n]$ exists over an algebraically closure of F. Considering Gröbner bases, $I = S$ if and only if there is a non-zero constant in a Gröbner basis, which we have shown is equivalent between the characteristic 0 and characteristic p cases for large enough p. Thus we have arrived at our result. □

Combining this result with those of the previous chapter, we have:

Theorem 5.2.6. *Existence of a sums-of-squares formula over an algebraically closed field is computable.*

Proof. For an algebraically closed field of finite characteristic p, we need only check the finitely many fields over $\mathbb{F}_p$ with degree given by 4.2.4. Thus there are finitely many possible coefficients, and this is computable.

For an algebraically closed field of characteristic 0, we check existence over an algebraically closed field of finite characteristic p for "large enough" p, as above. □

Unfortunately, the bounds are far to large for this to be practically computable.

5.3 A Computation

Using the Computer Algebra System SINGULAR, we can actually compute a Gröbner basis in the $[2, 2, 2]$ case. Explicitly, in the polynomial ring

$$\mathbb{Q}[x_{111}, x_{112}, x_{121}, x_{122}, x_{211}, x_{212}, x_{221}, x_{222}]$$

we compute a Gröbner basis for the ideal generated by the polynomials

$$x_{111}^2 + x_{211}^2 - 1$$
$$x_{112}^2 + x_{212}^2 - 1$$
$$x_{121}^2 + x_{221}^2 - 1$$
$$x_{122}^2 + x_{222}^2 - 1$$
$$x_{121}x_{111} + x_{221}x_{211}$$
$$x_{122}x_{112} + x_{222}x_{212}$$
$$x_{112}x_{111} + x_{212}x_{211}$$
$$x_{122}x_{121} + x_{222}x_{221}$$
$$x_{122}x_{111} + x_{121}x_{112} + x_{222}x_{211} + x_{221}x_{212}$$

using the degree reverse lexicographical ordering, we obtain the following Gröbner basis:

$$
\begin{array}{ll}
x_{221}^2 + x_{222}^2 - 1 & x_{211}x_{221} + x_{212}x_{222} \\
x_{121}x_{221} + x_{122}x_{222} & x_{112}x_{221} - x_{111}x_{222} \\
x_{212}^2 + x_{222}^2 - 1 & x_{211}x_{212} + x_{221}x_{222} \\
x_{122}x_{212} + x_{111}x_{221} & x_{121}x_{212} - x_{112}x_{221} \\
x_{112}x_{212} + x_{122}x_{222} & x_{111}x_{212} + x_{122}x_{221} \\
x_{211}^2 - x_{222}^2 & x_{122}x_{211} - x_{111}x_{222} \\
x_{121}x_{211} + x_{112}x_{222} & x_{112}x_{211} + x_{121}x_{222} \\
x_{111}x_{211} - x_{122}x_{222} & x_{122}^2 + x_{222}^2 - 1 \\
x_{121}x_{122} + x_{221}x_{222} & x_{112}x_{122} + x_{212}x_{222} \\
x_{111}x_{122} + x_{212}x_{221} & x_{121}^2 + x_{221}^2 - 1 \\
x_{112}x_{121} + x_{111}x_{122} + x_{212}x_{221} + x_{211}x_{222} & x_{111}x_{121} + x_{211}x_{221} \\
x_{112}^2 + x_{212}^2 - 1 & x_{111}x_{112} + x_{211}x_{212} \\
x_{111}^2 + x_{211}^2 - 1 & x_{212}x_{221}x_{222} - x_{211}x_{222}^2 + x_{211} \\
x_{122}x_{221}x_{222} - x_{121}x_{222}^2 + x_{121} & x_{111}x_{221}x_{222} + x_{112}x_{222}^2 - x_{112}
\end{array}
$$

Note that the maximum degree of a polynomial in this Gröbner basis is 3, far less than the bound of $2 \cdot 4^{2^{2 \cdot 2 \cdot 2 - 1}} \approx 2.3 \times 10^{77}$ that we used in our computations in the previous section.

Unfortunately, SINGULAR is unable to compute a Gröbner basis in any nontrivial cases beyond this. In the $[2, 2, 2]$ case, we already have a large number of variables and polynomials, and these increase exponentially as r, s, n increase.

5.4 Potential Improvements and Discussion

Although the bound we obtained in 5.2.5 is atrocious, there are significant opportunities to improve this bound. In the above computation, we saw that the actual degrees appearing in the Gröbner basis were significantly smaller than the bound from [5]. Tightening this degree bound would have a huge impact on the final bound we obtained, since it impacts both the number of steps in Buchberger's Algorithm and the bound on P in 5.2.3. Furthermore, in

order to avoid solving a very difficult recursion, in the proof of 5.2.4 we use the worst case degree bound at each step, even though we begin with polynomials that all have degree 2. A more careful analysis of the degrees at each step should provide a sharper bound.

By analyzing the coefficients that appear throughout Buchberger's Algorithm, we have given an explicit bound for p for which the characteristic 0 and characteristic p cases coincide over algebraically closed fields. Combining this with the degree bound from 4.2.4, we know that if there is a difference between the characteristic 0 and characteristic p cases, this can be detected in one of finitely many finite fields. In particular, we have shown that existence of a sums-of-squares formula over an algebraically closed field is theoretically computable, though not practically computable, since our bounds are so large. Improving the analysis of the coefficients could further restrict this search.

Analysis of the coefficients in Buchberger's Algorithm also has a small chance of supplying a proof of independence from the characteristic for algebraically closed fields, if it somehow turns out that the only coefficients that appear are ± 1.

CHAPTER 6

Group Action on the Variety of Sums-of-Squares Formulas

In this chapter, we use three equivalent perspectives on sums-of-squares formulas: the Hurwitz Matrix equations, a matrix of vectors, and our reformulation to view sums-of-squares formulas as a variety. Taking advantage of these different formulations, we define a group action of a product of orthogonal groups on this variety, which act by orthogonal change of variables. This is also an algebraic group action.

Using this group action, we establish a lower bound for the dimension of the variety of sums-of-squares formulas (when they exist) over a quadratically closed field, showing in particular that this variety is positive dimensional. This is done by considering a sums-of-squares formula as a matrix of vectors and showing that these vectors span $\mathbb{F}^n$. Then we can show that $O(n, F)$ acts freely, since the only matrix that fixes all of $\mathbb{F}^n$ is the identity matrix. From this, we conclude that the dimension of the variety of sums-of-squares formula is at least dim $O(n, F) = \frac{n(n-1)}{2}$. $O(r, F)$ and $O(s, F)$ also individually act freely, though the bounds they provide are smaller.

However, the entire action is not free in general. In order to prove this, we define a general notion of "orthonormal," and show that, over quadratically closed fields, orthonormal sets can always be extended to orthonormal bases.

We then compute the actual dimension of the variety in a few cases, which happens to match the bound we have found. However, it is very unlikely that this pattern will continue for larger $[r, s, n]$.

Finally, we discuss the additional questions raised by the existence of this group action.

6.1 First Perspective

The first perspective is the classical one, from which the Hurwitz Matrix Equations arise and the earliest results on sums-of-squares formulas were obtained using linear algebra.

Suppose we have a sums-of-squares formula

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots + y_s^2) = z_1^2 + \cdots + z_n^2$$

over a field F (of characteristic not 2), so z_k is bilinear in the x's and y's.

Writing $X = \begin{pmatrix} x_1 \\ \vdots \\ x_r \end{pmatrix}$, $Y = \begin{pmatrix} y_1 \\ \vdots \\ y_s \end{pmatrix}$, and $Z = \begin{pmatrix} z_1 \\ \vdots \\ z_n \end{pmatrix}$, a sums-of-squares formula is

$$X^T X \cdot Y^T Y = Z^T Z$$

such that Z is bilinear in X and Y.

Then $Z = AY$, where the entries of A are linear in X.

Thus a sums-of-squares formula is

$$X^T X \cdot Y^T Y = Y^T A^T A Y$$

such that the entries of A are linear in X.

We can then write $A = A_1 x_1 + \cdots + A_r x_r$, where each A_i is an $n \times s$ matrix with entries in F.

6.2 Second Perspective

Suppose we have sums-of-squares formula of type $[r, s, n]$ over a field F:

$$(x_1^2 + \cdots + x_r^2)(y_1^2 + \cdots + y_s^2) = z_1^2 + \cdots + z_n^2$$

where the z_i are bilinear expressions in the x's and y's.

From the reformulation in Chapter 3, existence of a sums-of-squares formula over F is equivalent to the existence of $\alpha_{ijk} \in F$ satisfying

$$\left\{ \begin{array}{ll} \sum_i \alpha_{ijk}^2 = 1 & \text{for all } j,k \\ \sum_i \alpha_{ijk}\alpha_{ij'k} = 0 & \text{for } j < j' \text{ and all } k \\ \sum_i \alpha_{ijk}\alpha_{ijk'} = 0 & \text{for all } j \text{ and } k < k' \\ \sum_i (\alpha_{ijk}\alpha_{ij'k'} + \alpha_{ijk'}\alpha_{ij'k}) = 0 & \text{for } j < j' \text{ and } k < k' \end{array} \right\}$$

We can then reformulate this as a condition on vectors.

For an arbitrary field F, we define a bilinear form on F^n by

$$(a_1, ..., a_n) \cdot (b_1, ..., b_n) = a_1b_1 + \cdots + a_nb_n$$

for $a_i, b_i \in F$. Of course, over $\mathbb{R}$, this is the usual inner product.

Now, writing the vector $v_{jk} = (\alpha_{1jk}, ..., \alpha_{njk})$, we can rewrite the polynomials above as:

$$\left\{ \begin{array}{ll} v_{jk} \cdot v_{jk} = 1 & \text{for all } j,k \\ v_{jk} \cdot v_{jk'} = 0 & \text{for all } j, k \neq k' \\ v_{jk} \cdot v_{j'k} = 0 & \text{for all } j \neq j', k \\ v_{jk} \cdot v_{j'k'} + v_{jk'} \cdot v_{j'k} = 0 & \text{for all } j \neq j', k \neq k' \end{array} \right\}$$

So we have that a sums of squares formula of type $[r, s, n]$ exists over F if and only if there are vectors $v_{jk} \in F^n$ for $1 \leq j \leq r$, $1 \leq k \leq s$ satisfying the above conditions.

We visualize this as an $r \times s$ matrix whose entries are unit vectors $v_{jk} \in F^n$, such that all rows and columns are orthogonal systems, and such that $v_{jk} \cdot v_{j'k'} = -v_{jk'} \cdot v_{j'k}$ for $j \neq j'$ and $k \neq k'$.

Note the similarity between this perspective and view sums-of-squares formulas over $\mathbb{Z}$ as consistently signed intercalacy matrices.

6.3 Third Perspective

The third perspective is the reformulation from Chapter 3; we saw that existence of a sums-of-squares formula over F is equivalent to asking if the set of polynomials

$$\left\{ \begin{array}{ll} \sum_i (x_{ijk}^2) - 1 & \text{for all } j,k \\ \sum_i x_{ijk}x_{ij'k} & \text{for } j < j' \text{ and all } k \\ \sum_i x_{ijk}x_{ijk'} & \text{for all } j \text{ and } k < k' \\ \sum_i (x_{ijk}x_{ij'k'} + x_{ijk'}x_{ij'k}) & \text{for } j < j' \text{ and } k < k' \end{array} \right\}$$

has a zero.

Let I be the ideal generated by this set of polynomials in $F[\{x_{ijk}\}]$.

Then the set of sums-of-squares formulas can be viewed as the zero set of the ideal I in $F[\{x_{ijk}\}]$.

We set the notation A_{rsn}^F for $F[\{x_{ijk}\}]/I$ and X_{rsn}^F for the variety of sums-of-squares formulas of type $[r,s,n]$ over F.

6.4 Defining a Group Action

Let $O(m,F)$ denote the orthogonal group of $m \times m$ matrices over F, so elements of $O(m,F)$ are matrices A such that $A^TA = \mathrm{Id}$.

In this section, we use the first perspective on sums-of-squares formulas to define an action of a product of orthogonal groups on the set of sums-of-squares formulas.

Proposition 6.4.1. *There is an action of $O(n,F)$ on the set of sums-of-squares formulas over F.*

Proof. Let $B \in O(n,F)$, so $B^TB = \mathrm{Id}$. Let

$$X^TX \cdot Y^TY = Z^TZ$$

be a sums of squares formula, $Z = AY$, with entries of A linear in X.

Let $Z' = BZ$, so $Z = B^{-1}Z'$. Then:

$$X^TX \cdot Y^TY = Z^TZ = (Z')^T(B^{-1})^TB^{-1}Z' = (Z')^TZ'$$

and $Z' = BZ = BAY$, and BA has entries linear in X, so this gives a sums-of-squares formula.

Thinking of this definition simply as $(B, A) \mapsto BA$, we see that it is clearly an action of $O(n, F)$ on the set of sums-of-squares formulas. □

Proposition 6.4.2. *There is an action of $O(s, F)$ on the set of sums-of-squares formulas over F.*

Proof. Let $C \in O(s, F)$, so $C^TC = \mathrm{Id}$. Let

$$X^TX \cdot Y^TY = Z^TZ$$

be a sums of squares formula, $Z = AY$, with entries of A linear in X.

Let $Y' = CY$, so $Y = C^{-1}Y'$. Then:

$$Z^TZ = X^TX \cdot Y^TY = X^TX \cdot (Y')^T(C^{-1})^TC^{-1}Y' = X^TX \cdot (Y')^TY'$$

and $Z = AY = AC^{-1}Y'$, and AC^{-1} has entries linear in Y, so this gives a sums-of-squares formula.

Thinking of this definition simply as $(C, A) \mapsto AC^{-1}$, we see that it is clearly an action of $O(s, F)$ on the set of sums-of-squares formulas. □

Proposition 6.4.3. *There is an action of $O(r, F)$ on the set of sums-of-squares formulas over F.*

Proof. Let $D \in O(r, F)$, so $D^TD = \mathrm{Id}$. Let

$$X^TX \cdot Y^TY = Z^TZ$$

be a sums of squares formula, $Z = AY$, with entries of A linear in X. Then we can write $A = A_1x_1 + \cdots + A_rx_r$. Formally, and for simplicity, we write this $A = (A_1 \ \cdots \ A_r) \cdot (x_1 \ \cdots \ x_r) = (A_1 \ \cdots \ A_r) \cdot X$.

Let $X' = DX$, so $X = D^{-1}X'$. Then:

$$Z^TZ = X^TX \cdot Y^TY = (X')^T(D^{-1})^TD^{-1}X' \cdot Y^TY = (X')^TX' \cdot Y^TY$$

and

$$Z = AY = ((A_1 \ \cdots \ A_r) \cdot X)Y = ((A_1 \ \cdots \ A_r) \cdot D^{-1}X')Y$$

and $D^{-1}X'$ is a vector linear in X', so $(A_1 \ \cdots \ A_r) \cdot D^{-1}X'$ is matrix linear in X'. So this is a sums-of-squares formula.

Thinking of this definition as $(D, (A_1 \ \cdots \ A_r) \cdot X) \mapsto (A_1 \ \cdots \ A_r) \cdot D^{-1}X$, we can see that this is an action of $O(r, F)$ on the set of sums-of-squares formulas. The details are worked out below.

Suppose $D_1, D_2 \in O(r, F)$. Then

$$(D_1D_2, (A_1 \ \cdots \ A_r) \cdot X) \mapsto (A_1 \ \cdots \ A_r) \cdot D_2^{-1}D_1^{-1}X$$

and

$$(D_2, (A_1 \ \cdots \ A_r) \cdot X) \mapsto (A_1 \ \cdots \ A_r) \cdot D_2^{-1}X$$

Write

$$D_2^{-1}X = \begin{pmatrix} d_{11}x_1 + \cdots + d_{1r}x_r \\ \vdots \\ d_{r1}x_1 + \cdots + d_{rr}x_r \end{pmatrix}$$

Then

$$\begin{aligned} (A_1 \ \cdots \ A_r) \cdot D_2^{-1}X &= (A_1 \ \cdot \ A_r) \cdot \begin{pmatrix} d_{11}x_1 + \cdots + d_{1r}x_r \\ \vdots \\ d_{r1}x_1 + \cdots + d_{rr}x_r \end{pmatrix} \\ &= A_1(d_{11}x_1 + \cdots + d_{1r}x_r) + \cdots + A_r(d_{r1}x_1 + \cdots + d_{rr}x_r) \\ &= (A_1d_{11} + \cdots + A_rd_{r1})x_1 + \cdots + (A_1d_{1r} + \cdots + A_rd_{rr})x_r \\ &= (A_1d_{11} + \cdots + A_rd_{r1} \ \cdots \ A_1d_{1r} + \cdots + A_rd_{rr}) \cdot X \end{aligned}$$

and then D_1 acting on this yields:

$$(A_1d_{11} + \cdots + A_rd_{r1} \ \cdots \ A_1d_{1r} + \cdots + A_rd_{rr}) \cdot D_1^{-1}X$$

Write $X' = D_1^{-1}X$, so this becomes::

$$\begin{aligned}(A_1 d_{11} + \cdots + A_r d_{r1} \ \cdots \ A_1 d_{1r} + \cdots + A_r d_{rr}) \cdot X' &= (A_1 \ \cdots \ A_r) \cdot D_2^{-1} X' \\ &= (A_1 \ \cdots \ A_r) \cdot D_2^{-1} D_1^{-1} X'\end{aligned}$$

Thus this is a group action on the set of sums-of-squares formulas. □

Proposition 6.4.4. *These actions commute, so we have an action of $O(r,F) \times O(s,F) \times O(n,F)$ on the set of sums-of-squares formulas over F.*

In fact, in the next section, we will see that we have an action of the group scheme $O(r,F) \times O(s,F) \times O(n,F)$ on the scheme X_{rsn}^F.

Proof. We first show that the actions of $O(s,F)$ and $O(n,F)$ commute. Let $B \in O(n,F)$, $C \in O(s,F)$, and

$$X^T X \cdot Y^T Y = Z^T Z$$

be a sums-of-squares formula, with $Z = AY$, entries of A linear in X.

Then

$$B \cdot C \cdot A = B \cdot (AC^{-1}) = BAC^{-1} = C \cdot BA = C \cdot B \cdot A$$

so the actions commute.

Next, we show that the actions of $O(s,F)$ and $O(r,F)$ commute. Let $D \in O(r,F)$, $C \in O(s,F)$, and

$$X^T X \cdot Y^T Y = Z^T Z$$

be a sums-of-squares formula, with $Z = AY$, and $A = (A_1 \ \cdots \ A_r) \cdot X$.

Then

$$\begin{aligned}C \cdot D \cdot A &= C \cdot D \cdot ((A_1 \ \cdots \ A_r) \cdot X) \\ &= C \cdot ((A_1 \ \cdots \ A_r) \cdot D^{-1} X) \\ &= (A_1 \ \cdots \ A_r) \cdot D^{-1} X) C^{-1} \\ &= (A_1 C^{-1} \ \cdots \ A_r C^{-1}) \cdot D^{-1} X \\ &= D \cdot (A_1 C^{-1} \ \cdots \ A_r C^{-1}) \cdot X \\ &= D \cdot ((A_1 \ \cdots \ A_r) \cdot X) C^{-1} \\ &= D \cdot C \cdot A\end{aligned}$$

so the actions commute.

Finally, we show that the actions of $O(r, F)$ and $O(n, F)$ commute. Let $D \in O(r, F)$, and $B \in O(n, F)$, and

$$X^T X \cdot Y^T Y = Z^T Z$$

be a sums-of-squares formula, with $Z = AY$, and $A = (A_1 \ \cdots \ A_r) \cdot X$.

Then

$$\begin{aligned} B \cdot D \cdot A &= B \cdot D \cdot (A(A_1 \ \cdots \ A_r) \cdot X) \\ &= B \cdot ((A_1 \ \cdots \ A_r) \cdot D^{-1} X) \\ &= B((A_1 \ \cdots \ A_r) \cdot D^{-1} X) \\ &= (BA_1 \ \cdots \ BA_r) \cdot D^{-1} X \\ &= D \cdot ((BA_1 \ \cdots \ BA_r) \cdot X) \\ &= D \cdot B((A_1 \ \cdots \ A_r) \cdot X) \\ &= D \cdot B \cdot A \end{aligned}$$

so the actions commute.

Thus we have an action of $O(r, F) \times O(s, F) \times O(n, F)$ on the set of sums-of-squares formulas. □

6.5 Group Action from the Second and Third Perspectives

Although the first perspective on sums-of-squares formulas made it easier to show that the action we defined was, in fact, an action, as we move forward, the second perspective will be easier to work with. We now note what this action looks like in the second perspective.

Note that $O(r, F)$ acts on sums-of-squares formula by change of X-coordinates, $O(s, F)$ by change of Y-coordinates, and $O(n, F)$ by change of Z-coordinates. Thus if we have a sums-of-squares formula given by

$$\begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix}$$

where $v_{ij} \in F^n$, we can see that the action is as follows.

$B \in O(n, F)$ acts by:

$$B \cdot \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} = \begin{pmatrix} Bv_{11} & \cdots & Bv_{1s} \\ \vdots & \ddots & \vdots \\ Bv_{r1} & \cdots & Bv_{rs} \end{pmatrix}$$

For $C = \begin{pmatrix} c_{11} & \cdots & c_{1s} \\ \vdots & \ddots & \vdots \\ c_{s1} & \cdots & c_{ss} \end{pmatrix} \in O(s, F)$, we have $C^{-1} = C^T$, and C acts by:

$$\begin{aligned} C \cdot \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} &= \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} C^{-1} \\ &= \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} \begin{pmatrix} c_{11} & \cdots & c_{s1} \\ \vdots & \ddots & \vdots \\ c_{1s} & \cdots & c_{ss} \end{pmatrix} \\ &= \begin{pmatrix} c_{11}v_{11} + \cdots + c_{s1}v_{1s} & \cdots & c_{1s}v_{11} + \cdots c_{ss}v_{1s} \\ \vdots & \ddots & \vdots \\ c_{11}v_{r1} + \cdots + c_{s1}v_{rs} & \cdots & c_{1s}v_{r1} + \cdots + c_{ss}v_{rs} \end{pmatrix} \end{aligned}$$

$D = \begin{pmatrix} d_{11} & \cdots & d_{1r} \\ \vdots & \ddots & \vdots \\ d_{r1} & \cdots & d_{rr} \end{pmatrix} \in O(r, F)$ acts by:

$$\begin{aligned} D \cdot \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} &= \begin{pmatrix} d_{11} & \cdots & d_{1r} \\ \vdots & \ddots & \vdots \\ d_{r1} & \cdots & d_{rr} \end{pmatrix} \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} \\ &= \begin{pmatrix} d_{11}v_{11} + \cdots + d_{1r}v_{r1} & \cdots & d_{11}v_{1s} + \cdots + d_{1r}v_{rs} \\ \vdots & \ddots & \vdots \\ d_{r1}v_{11} + \cdots + d_{rr}v_{r1} & \cdots & d_{r1}v_{1s} + \cdots + d_{rr}v_{rs} \end{pmatrix} \end{aligned}$$

From the third perspective, we can see that this is an action on the variety of sums-of-squares formulas. An element in $O(r,F) \times O(s,F) \times O(n,F)$ acts by a product of linear changes of variables, so this clearly induces an endomorphism f on $F[\{x_{ijk}\}]$. We have seen that this endomorphism takes sums-of-squares formulas to sums-of-squares formulas. By considering our reformulation, we can also see that it preserves the ideal I. Thus we have an automorphism of $F[\{x_{ijk}\}]/I$, and so an action of $O(r,F) \times O(s,F) \times O(n,F)$ on the variety of sums-of-squares formulas.

6.6 A Lower Bound on Dimension

We use this action (actually just the action of $O(n,F)$) to establish a lower bound for the dimension of the variety of sums-of-squares formula, when a sums-of-squares formula exists.

To show this, we first require a technical lemma, for which we use the second perspective of sums-of-squares formulas, viewing them as matrices with vector entries.

Lemma 6.6.1. *Let F be quadratically closed. Suppose we have a sums-of-squares formula of type $[r,s,n]$ given by $\{v_{ij}\}$, where n is minimal such that a sums-of-squares formula of type $[r,s,n]$ exists. Then the v_{ij} span F^n.*

Proof. Let V be the span of $\{v_{ij}\}$, and suppose V has dimension $m < n$.

Let $\phi_n(x_1,...,x_n) = x_1^2 + \cdots + x_n^2$, which is a quadratic form on F^n. Then $\phi_n|_V$ is a quadratic form on V.

Let $f : V \to F^m$ be a linear isomorphism. Then we get a quadratic form on F^m by taking $\psi_m(y) = \phi_n|_V(f^{-1}(y))$ for $y \in F^m$.

We claim that ψ_m is nondegenerate. We have that V is spanned by vectors $\{v_{ij}\}$ with $v_{ij} \cdot v_{ij} = 1$ for all i,j. Choose a maximal linearly independent subset of $\{v_{ij}\}$ (so a basis for V taken from the v_{ij}), and write it $w_1,...,w_m$. Note that we do not necessarily have $w_i \cdot w_j = 0$ for $i \neq j$. Suppose $a_1w_1 + \cdots + a_mw_m \in V$

Consider

$$\begin{pmatrix} f^{-1}(v_{11}) & \cdots & f^{-1}(v_{1s}) \\ \vdots & \ddots & \vdots \\ f^{-1}(v_{r1}) & \cdots & f^{-1}(v_{rs}) \end{pmatrix}$$

Our sums-of-squares formula gives us that $\phi_r(X)\phi_s(Y) = \phi_n|_V(Z)$ for Z bilinear in X and Y. Then:

$$\begin{aligned} \phi_r(X)\phi_s(Y) &= \phi_n|_V(Z) \\ &= \phi_n|_V(f^{-1}(f(Z))) \\ &= \psi_m(f(Z)) \end{aligned}$$

F is quadratically closed, so all nondegenerate quadratic forms over F are equivalent [4], and there is a linear transformation $M \in GL_m(F)$ such that $\psi_m = M \cdot \phi_m$, where M acts by $M \cdot \phi_m(W) = \phi_m(M^T W)$. Then:

$$\begin{aligned} \phi_r(X)\phi_s(Y) &= \psi_m(f(Z)) \\ &= M\phi_m(f^{-1}(Z)) \\ &= \phi_m(M^T f^{-1}(Z)) \end{aligned}$$

$M^T f^{-1}(Z)$ is bilinear in X and Y, so this gives a sums-of-squares formula of type $[r, s, m]$ over F with $m < n$, a contradiction. □

Theorem 6.6.2. *Let F be an quadratically closed field, and let n be minimal such that a sums-of-squares formula of type $[r, s, n]$ exists. Then $O(n, F)$ acts freely on the variety X of sums-of-squares formulas, and hence dim $(X) \geq \frac{n(n-1)}{2}$.*

Proof. Suppose $\{v_{ij}\}$ give a sums-of-squares formula that is fixed by $B \in O(n, F)$. Then $Bv_{ij} = v_{ij}$ for all i, j. Since the v_{ij} span F^n, this implies that B is the identity. Thus $O(n, F)$ acts freely on X.

dim $O(n, F) = \frac{n(n-1)}{2}$, so then dim $(X) \geq \frac{n(n-1)}{2}$. □

In particular, this means that when a sums-of-squares formula exists, the variety of sums-of-squares formula of that type has positive dimension, so there are many sums-of-squares formulas.

The actions of $O(r, F)$ and $O(s, F)$ are also free, and this is significantly easier to prove than for $O(n, F)$. However, since $r, s \leq n$ if a sums-of-squares formula exists, they provide a weaker bound on dimension.

Proposition 6.6.3. *The action of $O(r, F)$ is free.*

Proof. Let $B = \begin{pmatrix} b_{11} & \cdots & b_{1r} \\ \vdots & \ddots & \vdots \\ b_{r1} & \cdots & b_{rr} \end{pmatrix} \in O(r, F)$, and let

$$\begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix}$$

give a sums-of-squares formula, with $v_{ij} \in F^n$.

Suppose $B \cdot \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} = \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix}$, so this sums-of-squares formula is fixed by B.

Then

$$\begin{pmatrix} b_{11} & \cdots & b_{1r} \\ \vdots & \ddots & \vdots \\ b_{r1} & \cdots & b_{rr} \end{pmatrix} \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix} = \begin{pmatrix} v_{11} & \cdots & v_{1s} \\ \vdots & \ddots & \vdots \\ v_{r1} & \cdots & v_{rs} \end{pmatrix}$$

so $b_{11}v_{11} + \cdots + b_{1r}v_{r1} = v_{11}$. Then, since the v_{i1} are linearly independent (this is proved in the next section), $b_{11} = 1$, and $b_{12} = \cdots = b_{1r} = 0$.

Repeating this for the rest of the rows, we get that B is the identity.

Thus $O(r, F)$ acts freely on sums-of-squares formula. □

Proposition 6.6.4. *The action of $O(s, F)$ is free.*

Proof. The proof is essentially the same as that for $O(r, F)$. □

However, the entire group action of $O(r) \times O(s) \times O(n)$ is not free in general, as we see in 6.8.

6.7 Extending Orthonormal Sets in Finite Characteristic

In this section, we give a generalization of "orthonormal" which is useful for our purposes, and we prove that we can extend orthonormal sets to orthonormal bases. This will be useful in the next section, where we consider whether or not our group action as a whole is free.

Let F be a quadratically closed field. We define an *orthonormal set* of F^n to be a set of vectors $\{v_i\}$ such that $v_i \cdot v_i = 1$ for all i and $v_i \cdot v_j = 0$ for all $i \neq j$. An *orthonormal basis* is an orthonormal set that spans F^n.

Note that we do not know a priori that an orthonormal set is linearly independent, because in characteristic p, we may have linearly dependent v and w such that $v \cdot w = 0$. However, this is true, and we prove it as a first lemma.

Lemma 6.7.1. *Let $\{v_i\}$ be an orthonormal set. Then $\{v_i\}$ is linearly independent.*

Proof. Suppose $a_1v_1 + \cdots a_nv_n = 0$ is a dependence relation. Then $(a_1v_1 + \cdots + a_nv_n) \cdot v_i = 0$ for all i, hence $0 = a_1v_1 \cdot v_i + \cdots a_nv_n \cdot v_n = a_i$ for all i. Thus the set is linearly independent. □

Assume we have an orthonormal set $\{v_1, ..., v_r\}$ in F^n. In the following two lemmas, we show that this can be extended to an orthonormal basis of F^n.

Lemma 6.7.2. *Suppose $r \leq n-2$. Then there is $v_{r+1} \in F^n$ such that $\{v_1, ..., v_r, v_{r+1}\}$ is an orthonormal set.*

Proof. Consider the set $W = \{w \in F^n \mid w \cdot v_i = 0 \text{ for all } v_i\}$. The v_i are linearly independent, so W is a subspace of F^n of dimension $n - r \geq 2$. Also note that since $v_i \cdot v_i = 1$ for all i and $v_i \cdot v_j = 0$ for all $i \neq j$, every vector in W is linearly independent from $\{v_i\}$.

If there is $w \in W$ with $w \cdot w \neq 0$, choose a square root of $w \cdot w$, and let $v_{r+1} = \frac{w}{\sqrt{w \cdot w}}$.

Then $\{v_1, ..., v_r, v_{r+1}\}$ is linearly independent, $v_{r+1} \cdot v_{r+1} = 1$, and $v_{r+1} \cdot v_j = 0$ for $j \neq r+1$. Thus $\{v_1, .., v_r, v_{r+1}\}$ is a linearly independent set.

Now, suppose for all vectors $w \in W$, we have $w \cdot w = 0$. Let $\{w_1, ..., w_{n-r}\}$ be any basis for W. Then $\{v_1, ..., v_r, w_1, ..., w_{n-r}\}$ is a basis for F^n. $w_1 \cdot v_i = 0$ for all i, and $w_1 \cdot w_1 = 0$. But this is a nondegenerate quadratic form, so we must have $w_1 \cdot w_i \neq 0$ for some i. Consider $w_1 + w_i$. $w_1 + w_i \in W$, and $(w_1 + w_i) \cdot (w_1 + w_i) = w_1 \cdot w_1 + 2w_1 \cdot w_i + w_i \cdot w_i = 2w_1 \cdot w_i \neq 0$, a contradiction.

Thus the orthonormal set can be extended. □

The following lemma completes the proof that orthonormal sets can be extended.

Lemma 6.7.3. *Suppose $r = n - 1$. Then there is $v_n \in F^n$ such that $\{v_1, ..., v_{n-1}, v_n\}$ is an orthonormal set (hence an orthonormal basis).*

Proof. Once again, let $W = \{w \in F^n \mid w \cdot v_i = 0 \text{ for all } v_i\}$. This is nontrivial, so take nonzero $w \in W$.

If $w \cdot w \neq 0$, we are done as in the previous proof.

If $w \cdot w = 0$, then, since $v_1, ..., v_{n-1}, w$ are linearly independent and hence span F^n, we have $w \cdot v = 0$ for all $v \in F^n$. This contradicts that the bilinear form is nondegenerate.

Thus the orthonormal set can be extended. □

Thus we have shown that we can always extend an orthonormal set to an orthonormal basis over a quadratically closed field, which will be used in the next section.

Note that if we do not have a quadratically closed field, we may not be able to extend every orthonormal set. For example, in $\mathbb{F}_3^4$, the vector $(1\,1\,1\,1)$ cannot be extended to an orthonormal basis.

6.8 A Special Case

Assume F is a quadratically closed field.

Although the individual actions of $O(r, F)$, $O(s, F)$, and $O(n, F)$ are free, the entire action of $O(r, F) \times O(s, F) \times O(n, F)$ may not be.

Consider the case $s = 1$. Here a sums-of-squares formula is essentially just an orthonormal set in F^n. Let $\{v_i\}$ be a sums-of-squares formula of type $[r, 1, n]$ (omitting the index for s since $s = 1$), and extend this to an orthonormal basis $v_1, ..., v_r, w_1, ..., w_{n-r}$ for F^n. Let $D \in O(r, F)$. We can write $(D \cdot \{v_i\})_j = \sum d_{ij} v_i$. Define B by $v_i \mapsto \sum d_{ij} v_i$ and $w_i \mapsto w_i$. This is easily seen to by in $O(n, F)$, and we have $B \cdot \{v_i\} = D \cdots \{v_i\}$. Thus we see that the action is not free.

However, the action is faithful. We may assume that $n \geq 2$. Let $w_1 = v_2$, $w_2 = v_1$, and $w_i = v_i$ for $i > 2$. Assume that $B \cdot \{w_i\} = D \cdot \{w_i\}$. Then for all i, we must have

$$d_{i1} v_1 + d_{i2} v_2 + \cdots = B v_i = d_{i1} v_2 + d_{i2} v_1 + \cdots$$

so $d_{i1} = d_{i2}$ for all i. But then D is not invertible, a contradiction. Thus we see the action is faithful.

We do not yet know if the action is faithful in general.

6.9 Comparing Lower Bound to Actual Dimension in Small Cases

Using the Computer Algebra System Macaulay, we can compute the dimension of X^F_{rsn} over various fields F. Over $\mathbb{C}$, $\mathbb{F}_3$, $\mathbb{F}_5$, and $\mathbb{F}_7$ (and likely all finite fields), X^F_{222} has dimension 1. This matches the lower bound provided by 6.6.2.

Furthermore, in the $[n, 1, n]$ case, a sums-of-squares formula is just an orthonormal basis over $\mathbb{F}$, thus the dimension is equal to the dimension of $O(n, F)$ and matches the lower bound provided by 6.6.2.

Of course, there is no reason to expect that the actual dimension will match the bound in larger cases. Unfortunately, Macaulay is unable to handle cases larger than $[2, 2, 2]$.

6.10 New Questions

The existence of this algebraic group action raises many new questions that can tell us more about the structure of the variety of sums-of-squares formulas, such as:

- Is the action faithful?
- What are the orbits? Is the action transitive? Is the variety of sums-of-squares formulas homogeneous?
- What is the stabilizer of the action?
- What are some invariants of the group action? Can we find a useful invariant theory?

Answers to these questions not only provide information about the structure of the action and variety, but can also provide insight into the question of whether existence of formulas is independent of the base field. For example, if every formula were in the same orbit as a formula that can be lifted to the integers, then existence of a sums-of-squares formula would be independent of the base field.

CHAPTER 7

Computer Searches

In this chapter, we discuss methods of conducting computer searches for sums-of-squares formulas over the integers and over finite fields. By conducting these searches over finite fields, we can look for formulas that do not come from formulas over the integers (existence of such a formula is unknown), and investigate the orbits of the group action introduced in Chapter 6. These searches are of particular interest since we proved in Chapter 5 that existence of a sums-of-squares formula over an algebraically closed field is computable.

We begin by giving a basic algorithm for finding sums-of-squares formulas over the integers. This is possible by using Yuzvinsky's characterization of sums-of-squares formulas using consistently signed intercalate matrices, and works by building such a matrix entry by entry.

Next, we discuss how the search can be refined, and provide an improved algorithm for finding sums-of-squares formulas over the integers. These refinements are done by reducing the number of possibilities we check at each entry, thus there are fewer matrices to check.

Then, we give an algorithm for finding sums-of-squares formulas over finite fields. This algorithm also works by filling in a formula entry by entry, however now the entries are unit vectors over the finite field. By running this search over $\mathbb{F}_3$, we can search for new formulas, and we find a formula that does not lift to the integers. Existence of such a formula was previously unknown. This formula is, however, related to a formula over the integers: it turns out that it is in the same orbit as a formula that can be lifted to the integers.

Finally, we discuss the constraints and issues that arise while trying to conduct computer searches for sums-of-squares formulas.

7.1 Basic Algorithm over the Integers

In this section, we introduce an algorithm for finding sums-of-squares formulas over the integers, based on Yuzvinky's equivalent formulation of sums-of-squares formulas over the integers using consistently signed intercalate matrices.

Recall that Yuzvinsky introduced consistently signed intercalate matrices, and showed that existence of a consistently signed intercalate matrix of type (r, s, n) is equivalent to existence of a sums-of-squares formula of type $[r, s, n]$ over the integers.

Definition 7.1.1. Suppose M is an $r \times s$ matrix with entries taken from a finite set of "colors." Let M_{ij} be the (i, j)-th entry of M.

M is *intercalate* if:

- The colors along each row (resp. column) are distinct.
- If $M_{ij} = M_{i'j'}$, then $M_{ij'} = M_{i'j}$. (Every 2×2 submatrix involves an even number of distinct colors.)

An intercalate matrix has *type* (r, s, n) if it is an $r \times s$ matrix with at most n colors.

An intercalate matrix M is *consistently signed* if there exist $\epsilon_{ij} = \pm 1$ such that

$$\epsilon_{ij}\epsilon_{ij'}\epsilon_{i'j}\epsilon_{i'j'} = -1$$

whenever $M_{ij} = M_{i'j'}$. (Every 2×2 submatrix with only two distinct colors must have an odd number of minus signs.)

Thus we can search for sums-of-squares formulas over the integers by searching for consistently signed intercalate matrices, and this is the approach we take. We take our set of "colors" to be $\{1, ..., n\}$.

The algorithm works by going through one entry (the (i, j)th entry) of the matrix at a time. The possible values for each entry are $\{-1, -2, ..., -n, +1, +2, ..., +n\}$, though we handle the color and the sign in separate matrices to improve efficiency of the search. For

the current value, we check this against previous entries, the (i', j')th entries with $i' \leq i$ and $j' \leq j$ to verify that the conditions for a consistently signed intercalate matrix are met. If they are met, we move to the next entry and try its first value. If one of the conditions fails, we move to the next value. If we have run out of values for the current entry, we reset the current value and go back to the previous entry and move to the next value there. Thus we ensure that all possibilities are checked, while avoiding some potentially unnecessary computations.

When coded in C++, this algorithm handles up through the $n = 12$ case very quickly when formulas exist, but begins to become impractically slow investigating cases beyond this. It is also very slow at determining that a formula doesn't exist; beyond $n = 5$ it starts to become very slow when a formula doesn't exist.

Over the next few pages, we give pseudo-code for this algorithm for finding sums-of squares formulas of type $[r, s, n]$ over the integers (by finding a consistently signed intercalate matrix).

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers

colors is an $r \times s$ matrix of integers, all initialized to 0

signs is an $r \times s$ matrix of values ± 1, all initialized to -1.

exists $\leftarrow$true

$i \leftarrow 0$

$j \leftarrow 0$

trigger $\leftarrow$true

colors will contain only the colors of the consistently signed intercalate matrix, and *signs* will only contain the signs. Separating them makes checking the row, column, and intercalacy conditions much quicker.

exists will ultimately indicate whether a formula exists or not. i and j will run through the entries in the matrices. *trigger* will be used to indicate that one of the row, column, or intercalacy conditions has failed for an entry.

In the next part, we give the algorithm for moving to the next value to test. If we have not run out of colors at the current entry, we simply go to the next color. Otherwise, if the sign is -1, we change the sign to $+1$ and start the colors over at 1. If we have exhausted all possibilities at the current entry, we reset the current entry and move back to the previous, starting the loop over. If there is no previous entry, we have exhausted all possibilities, so no formula exists and we end the search.

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers (cont'd)

```
while 0 ≤ i < r, 0 ≤ j < s do
    if colors[i][j] < n then                          ▷ Increment current color
        colors[i][j] ← colors[i][j] + 1
    else if signs[i][j] = −1 then                     ▷ If current color is already maximum,
        colors[i][j] ← 1                              ▷ reset to 0 and change sign to +1
        signs[i][j] ← 1
    else                                              ▷ If current color is maximum and sign is positive,
        colors[i][j] ← 0
        signs[i][j] ← −1
        if j > 0 then                                 ▷ go back to previous entry
            j ← j − 1
        else
            i ← i − 1
            j ← s − 1
        if i < 0 then                                 ▷ if there is no previous entry,
            exists ← false                            ▷ no formula exists and we are done
            Break
        Continue
    trigger ←true
```

Next, we check the row condition that all colors in a row must be distinct, by checking the current entry against all previous entries in the same row. We are still within the while

loop. If this condition fails, we move to the next iteration of the while loop in order to check the next possible value.

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers (cont'd)

```
for 0 ≤ j' < j do                                   ▷ check row condition
    if colors[i][j] = colors[i][j'] then
        trigger ←false
        Break
if not trigger then
    Continue
```

Still in the while loop, we check the column condition that all colors, checking that the current color is not equal to any of the previous colors in the column. If this fails, we move to the next iteration of the while loop.

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers (cont'd)

```
for 0 ≤ i' < i do                                   ▷ check column condition
    if colors[i][j] = colors[i'][j] then
        trigger ←false
        Break
if not trigger then
    Continue
```

Still in the while loop, we check the intercalacy condition that every 2×2 submatrix has an even number of colors and an odd number of signs. This means that the (i, j)th and (i', j')th color are the same if and only if the (i, j')th and (i', j)th color are the same, and that if the (i, j)th and (i', j')th entry have the same sign if and only if the (i, j')th and (i', j)th entry have different signs.

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers (cont'd)

for $0 \leq i' < i,\ 0 \leq j' < j$ **do** ▷ check intercalacy condition
  if $colors[i][j] = colors[i'][j']$ **then**
    if $colors[i][j'] \neq colors[i'][j]$ **then**
      $trigger \leftarrow$false
      Break
    if $signs[i][j] = signs[i'][j']$ and $signs[i'][j] = signs[i][j']$ **then**
      $trigger \leftarrow$false
      Break
    if $signs[i][j] \neq signs[i'][j']$ and $signs[i'][j] \neq signs[i][j']$ **then**
      $trigger \leftarrow$false
      Break
  else if $colors[i][j'] = colors[i'][j]$ **then**
    $trigger \leftarrow$false
    Break

At this point, still in the while loop, if *trigger* is true, then all conditions have passed, so we go to the next entry in the matrix and start the next iteration of the loop there.

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers (cont'd)

if *trigger* **then** ▷ If none of the conditions have failed,
  if $j < s - 1$ **then** ▷ go to next entry
    $j \leftarrow j + 1$
  else
    $i \leftarrow i + 1$
    $j \leftarrow 0$

At the beginning of the next part, we have left the while loop. If *exists* is true, then a formula has been completely filled in and we print that formula. If *exists* is false, then no there is no formula over the integers of the given type.

Algorithm 1 Existence of Sums-of-Squares Formulas over the Integers (cont'd)

```
if exists then                ▷ If it exists, print the consistently signed intercalate matrix
    print "A formula is given by:"
    print newline
    for 0 ≤ i < r do
        for 0 ≤ j < s do
            if signs[i][j] = 1 then
                print "+"
            else
                print "-"
            print colors[i][j]
            print " "
        print newline
else
    print "No such formula exists."
```

For example, running this algorithm for $[4, 9, 12]$ produces the consistently signed intercalacy matrix:

$$\begin{pmatrix} -1 & -2 & -3 & -4 & -5 & -6 & -7 & -8 & -9 \\ -2 & +1 & -4 & +3 & -6 & +5 & -8 & +7 & -10 \\ -3 & +4 & +1 & -2 & -7 & +8 & +5 & -6 & -11 \\ -4 & -3 & +2 & +1 & -8 & -7 & +6 & +5 & -12 \end{pmatrix}$$

This corresponds to the following sums-of-squares formula of type $[4, 9, 12]$:

$$
\begin{aligned}
z_1 &= -x_1y_1 + x_2y_2 + x_3y_3 + x_4y_4 \\
z_2 &= -x_1y_2 - x_2y_1 - x_3y_4 + x_4y_3 \\
z_3 &= -x_1y_3 + x_2y_4 - x_3y_1 - x_4y_2 \\
z_4 &= -x_1y_4 - x_2y_3 + x_3y_2 - x_4y_1 \\
z_5 &= -x_1y_5 + x_2y_6 + x_3y_7 + x_4y_8 \\
z_6 &= -x_1y_6 - x_2y_5 - x_3y_8 + x_4y_7 \\
z_7 &= -x_1y_7 + x_2y_8 - x_3y_5 - x_4y_6 \\
z_8 &= -x_1y_8 - x_1y_7 + x_3y_6 - x_4y_5 \\
z_9 &= -x_1y_9 \\
z_{10} &= -x_2y_9 \\
z_{11} &= -x_3y_9 \\
z_{12} &= -x_4y_9
\end{aligned}
$$

7.2 Improved Algorithm over the Integers

In this section, we give an improved algorithm for searching for sums-of-squares formulas over the integers. This is done by observing through permutations and counting arguments that if a sums-of-squares formula exists, then one exists that has a particular form.

Assume that we have a sums-of-squares formula of type $[r, s, n]$ over the integers, given by a consistently signed intercalate matrix.

First, observe that by renumbering the colors, we can assume that the first row is

$$1 \quad 2 \quad 3 \quad \cdots \quad s$$

Furthermore, we have n colors and rs entries in the matrix. This means that some color must occur at least $\lceil \frac{rs}{n} \rceil$ times. By renumbering and permutation, we can assume that this

color is 1, and these occur along the diagonal (we will deal with signs separately):

$$\begin{array}{cccccc} 1 & 2 & \cdots & \lceil \frac{rs}{n} \rceil & \cdots & s \\ & 1 & & & & \\ & & \ddots & & & \\ & & & 1 & & \end{array}$$

Then, by the intercalacy condition, we know what the first $\lceil \frac{rs}{n} \rceil$ colors in the first column must be:

$$\begin{array}{cccccc} 1 & 2 & 3 & \cdots & \lceil \frac{rs}{n} \rceil & \cdots & s \\ 2 & 1 & & & & & \\ 3 & & 1 & & & & \\ \vdots & & & \ddots & & & \\ \lceil \frac{rs}{n} \rceil & & & & 1 & & \end{array}$$

Finally, having these values filled eliminates values in the remaining entries:

$$\begin{array}{ccccccc} 1 & 2 & 3 & \cdots & \lceil \frac{rs}{n} \rceil & \cdots & s \\ 2 & 1 & \{\neq 1,2,3\} & & \{\neq 1,2,\lceil \frac{rs}{n} \rceil\} & & \{\neq 1,2,s\} \\ 3 & \{\neq 1,2,3\} & 1 & & & & \\ \vdots & & & \ddots & & & \\ \lceil \frac{rs}{n} \rceil & \{\neq 1,2,\lceil \frac{rs}{n} \rceil\} & & & 1 & & \\ \{\neq 1,...,\lceil \frac{rs}{n} \rceil\} & \{\neq 1,2\} & \{\neq 1,3\} & & & & \\ \vdots & & & & & & \end{array}$$

These observations are implemented in the following algorithm.

As in the previous section, we split the consistently signed intercalate matrix into colors and signs. However, we would now like to consider only specific color possibilities for each entry, and these differ from entry to entry. To handle this, *colors* is now an $r \times s$ matrix of pointers, and the (i, j)th entry points to an array of the possible colors for that entry (as listed above). We create an additional matrix, *number*, which tracks which possibility we are currently on, as well as how many possibilities there are for each entry.

When implemented in C++, this algorithm actually performs slightly worse at finding formulas, handing cases up through $n = 11$ quickly and slowing dramatically beyond this. This is likely a result of the extra overhead required to organize the possibilities; it's theoretically an improvement because we have fewer possibilities to check. Also, the order that we checked formulas in the previous algorithm would find formulas with this form fairly early in the search. However, this algorithm is significantly better at showing that a formula doesn't exist; it can handle these cases up through $n = 9$. Here narrowing down the possibilities makes a dramatic difference.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved)

colors is an $r \times s$ matrix of pointers

signs is an $r \times s$ matrix of values ± 1, all initialized to 1

number is an $r \times s$ matrix, each entry has two values: *current* and *max*

Now, we begin to make the list of possibilities for each entry, and adjusting $number[i][j]$ as necessary.

We start by filling in the first row with $1, 2, 3, ...$

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

for $0 \leq i < r,\ 0 \leq j < s$ **do**

  if $i = 0$ **then**

    $colors[i][j]$ is an array with one element

    $colors[i][j][0] \leftarrow j + 1$

    $number[i][j].current \leftarrow 0$

    $number[i][j].max \leftarrow 1$

Still in the for loop, we now fill in what we know of the first column.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

else if $j = 0$ and $0 < i < \left\lceil \frac{rs}{n} \right\rceil$ **then**

$colors[i][j]$ is an array with one element

$colors[i][j][0] \leftarrow i + 1$

$number[i][j].current \leftarrow -1$

$number[i][j].max \leftarrow 1$

Still in the for loop, we fill in what we know of the diagonal.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

else if $i = j$ and $0 < i < \left\lceil \frac{rs}{n} \right\rceil$ **then**

$colors[i][j]$ is an array with one element

$colors[i][j][0] \leftarrow 1$

$number[i][j].current \leftarrow -1$

$number[i][j].max \leftarrow 1$

Still in the for loop, we fill in the possibilities for the first $\left\lceil \frac{rs}{n} \right\rceil$ rows.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

else if $i < \left\lceil \frac{rs}{n} \right\rceil$ **then**

$colors[i][j]$ is an array with $n - 3$ elements

$k \leftarrow 0$

$m \leftarrow 0$

while $m < n$ **do**

if $m \neq 0, i, j$ **then**

$colors[i][j][k] \leftarrow m + 1$

$k \leftarrow k + 1$

$m \leftarrow m + 1$

$number[i][j].current \leftarrow -1$

$number[i][j].max \leftarrow n - 3$

Still in the for loop, we fill in the rest of the first column.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

else if $j = 0$ **then**

$colors[i][j]$ is an array with $n - \left\lceil \frac{rs}{n} \right\rceil$ elements

$m \leftarrow \left\lceil \frac{rs}{n} \right\rceil$

$k \leftarrow 0$

while $m < n$ **do**

$colors[i][j][k] \leftarrow m + 1$

$k \leftarrow k + 1$

$m \leftarrow m + 1$

$number[i][j].current \leftarrow -1$

$number[i][j].max \leftarrow n - \left\lceil \frac{rs}{n} \right\rceil$

Still in the for loop, we fill in the remaining entries in the first $\left\lceil \frac{rs}{n} \right\rceil$ columns.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

else if $j < \left\lceil \frac{rs}{n} \right\rceil$ **then**

$colors[i][j]$ is an array with $n - 2$ elements

$k \leftarrow 0$

$m \leftarrow 0$

while $m < n$ **do**

if $m \neq 0, j$ **then**

$colors[i][j][k] \leftarrow m + 1$

$k \leftarrow k + 1$

$m \leftarrow m + 1$

$number[i][j].current \leftarrow -1$

$number[i][j].max \leftarrow n - 2$

Finally, still in the for loop, we fill in the remaining entries:

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

else

$colors[i][j]$ is an array with $n-1$ elements

$k \leftarrow 0$

$m \leftarrow 0$

while $m < n$ **do**

if $m \neq j$ **then**

$colors[i][j][k] \leftarrow m+1$

$k \leftarrow k+1$

$m \leftarrow m+1$

$number[i][j].current \leftarrow -1$

$number[i][j].max \leftarrow n-1$

Now, with all of the possibilities for each entry set up, we begin to check entry by entry to build a consistently signed intercalate matrix.

We begin by incrementing the current value. If we can just go to the next possible color, we do that. If we have run out of colors but the sign is still 1, we set the sign to -1 and reset the color to the first possibility. If we have run out of possibilities, we go back to the previous entry. If there is no previous entry to change (note that the first row cannot be changed), there is no formula and we end the search.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

```
while 1 ≤ i < r and 0 ≤ j < s do
    if number[i][j].current < number[i][j].max − 1 then
        number[i][j].current ← number[i][j].current + 1
    else if signs[i][j] = 1 then
        number[i][j].current ← 0
        signs[i][j] ← −1
    else
        number[i][j].current ← −1
        signs[i][j] ← 1
        if j > 0 then
            j ← j − 1
        else
            i ← i − 1
            j ← s − 1
        if i < 1 then
            exists ←false
            Break
        Continue
```

Still in the while loop, we now check the row condition. If it fails at any point, we move to the next iteration of the loop.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

trigger ←true

for $0 \leq j' < j$ **do**

if $colors[i][j][number[i][j].current] = colors[i][j'][number[i][j'].current]$ **then**

trigger ←false

Break

if not *trigger* **then**

Continue

Still in the while loop, we check the column condition. If it fails, we move to the next iteration of the loop.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

for $0 \leq i' < i$ **do**

if $colors[i][j][number[i][j].current] = colors[i'][j][number[i'][j].current]$ **then**

trigger ←false

Break

if not *trigger* **then**

Continue

Still in the while loop, we check the intercalacy condition. If it fails, we move to the next iteration of the loop.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

for $0 \leq i' < i,\ 0 \leq j' < j$ **do**

if $colors[i][j][number[i][j].current] = colors[i'][j'][number[i'][j'].current]$ **then**

if $colors[i][j'][number[i][j'].current] \neq colors[i'][j][number[i'][j].current]$ **then**

$trigger \leftarrow$false

Break

if $signs[i][j] = signs[i'][j']$ and $signs[i][j'] = signs[i'][j]$ **then**

$trigger \leftarrow$false

Break

if $signs[i][j] \neq signs[i'][j']$ and $signs[i][j'] \neq signs[i'][j]$ **then**

$trigger \leftarrow$false

Break

else if $colors[i][j'][number[i][j'].current] = colors[i'][j][number[i'][j].current]$ **then**

$trigger \leftarrow$false

Break

Still in the while loop, if all the conditions have passed, we move to the next entry.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

if $trigger$ **then**

if $j < s - 1$ **then**

$j \leftarrow j + 1$

else

$i \leftarrow i + 1$

$j \leftarrow 0$

Now outside of the while loop, if a formula exists, we print it. If it doesn't exist, we indicate this.

Algorithm 2 Existence of Sums-of-Squares Formulas over the Integers (Improved, cont'd)

```
if exists then
    print "A formula is given by:"
    print newline
    for 0 ≤ i < r do
        for 0 ≤ j < s do
            if signs[i][j] = 1 then
                print "+"
            else
                print "-"
            print colors[i][j][number[i][j].current]
            print " "
        print newline
else
    print "No such formula exists."
```

7.3 Algorithm over Finite Fields

In this section, we introduce an algorithm for finding sums-of-squares over finite fields. This algorithm can be used to search for sums-of-squares formulas over $\mathbb{F}_3$ which do not come from sums-of-squares formulas over the integers.

The approach is similar to the algorithm over the integers, except now we must visualize an $r \times s$ matrix of vectors rather than colors. We follow the second perspective on sums-of-squares formulas, introduced in section 6.2. We begin by listing all vectors of length 1 in $\mathbb{F}_3^n$ (meaning $v_1^2 + v_2^2 + \cdots + v_n^2 = 1$), and this list takes the place of "colors" over the integers. We then work entry by entry as before, checking the row, column, and intercalacy conditions along the way.

In the algorithm below, we assume that if a sums-of-squares formula exists, then one can

be found where the first row consists of basic unit vectors. This is true for quadratically closed fields as a consequence of 6.7, but might not be true for finite fields. Thus, the algorithm as presented cannot definitively say that a formula does not exist. If one would like to be able to say that a formula does not exist, one can easily adjust this algorithm to not fix the first row as basic unit vectors. The benefit of making this assumption is a gain in efficiency.

In the pseudo-code beginning on the next page, operations are modulo p. The pseudo-code is written for a search over $\mathbb{F}_p$ where p is prime, but this approach can easily be adapted for arbitrary finite fields. This was implemented in C++ for $\mathbb{F}_3$ by representing $0, 1, 2$ in binary using booleans. Operations were defined element by element, rather than relying on the usual algorithms. This resulted in a significant gain in computability; relying on normal addition, cases beyond $n = 4$ became very slow. Implemented with the booleans, this algorithm becomes slow past the $[6, 7, 8]$ case. It is very slow at determining that a formula does not exist.

The first step in this algorithm is to build a set of all vectors in $\mathbb{F}_p^n$ of length 1, which will be entries in an $r \times s$ matrix that gives a sums-of-squares formula. v will run through all (finitely many vectors) in $\mathbb{F}_p^n$, and the vectors of length 1 will be added to $ourSet$. Counter will track how many vectors we have tested, relative to the total number of vectors in $\mathbb{F}_p^n$, which is p^n.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field

v is a vector of integers modulo p, all initialized to 0

$ourSet$ is a collection (vector) of vectors of integers modulo p

$counter \leftarrow 0$

For each vector in $\mathbb{F}_p^n$, we check if it has length 1. If it does, we add it to $ourSet$. Then we go to the next vector, and we keep count of how many vectors we have checked.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

while $counter < p^n$ **do** ▷ In this loop, we add all unit vectors to $ourSet$

  $sum \leftarrow 0$

  for $0 \leq i < n$ **do**

    $sum \leftarrow sum + v[i]^2$

  if $sum = 1$ **then**

    $ourSet.pushback(v)$

  for $0 \leq i < n$ **do**

    if $v[i] < p - 1$ **then**

      $v[i] \leftarrow v[i] + 1$

      Break

    $v[i] \leftarrow 0$

  $counter \leftarrow counter + 1$

We now build a set of standard unit vectors, which will be used to build the first row of our sums-of-squares formula. e will contain the standard unit vectors, and we add each standard unit vector to e.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

w is a vector of integers modulo p, all initialized to 0

e is a collection (vector) of vectors of integers modulo p

for $0 \leq i < n$ **do** ▷ We build e as the set of basic unit vectors

  $w[i] \leftarrow 1$

  $e.pushback(w)$

  $w[i] \leftarrow 0$

Now, we begin the process of building a sums-of-squares formula. $formula$ is an $r \times s$ matrix of n-dimensional vectors, and this will be our sums-of-squares formula (if it exists). $number$ is an $r \times s$ matrix of integers, and it keeps track of which index in the set of all unit vectors the current entry is on. This enables us to iterate through all unit vectors.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

$formula$ is an $r \times s$ matrix of vectors of integers modulo p

$number$ is an $r \times s$ matrix of integers, all initialized to -1.

If we change some of the entries in $number$ here, we can find different sums-of-squares formulas. For example, in the next section, we consider a formula that is found by adding the line $number[1][0] \leftarrow 100$.

Now, we fill in the first row with the first s standard unit vectors.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

▷ Adding $number[1][0] \leftarrow ?$ here can allow us to find different formulas

for $0 \leq i < s$ **do** ▷ The first row is basic unit vectors

  $formula[0][i] \leftarrow e[i]$

We now begin building the rest of our formula. At each entry, we will run through all the possibilities (i.e., all the unit vectors) using $number[i][j]$ to keep track of which one we are on, and updating $formula[i][j]$ at each step.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

```
i ← 1
j ← 0
while 1 ≤ i < r, 0 ≤ j < s do
    while number[i][j] + 1 < ourSet.size() do
        number[i][j] ← number[i][j] + 1
        formula[i][j] ← ourSet[number[i][j]]
```

Inside of both while loops, we check the column condition for the current entry, checking that the dot product of the current entry with previous entries in the same column is 0. If any of these fail, we go to the next iteration of the loop in order to check the next value.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

```
        for 0 ≤ i' < i do                          ▷ Check column condition
            sum ← 0
            for 0 ≤ k < n do
                sum ← sum + formula[i'][j][k] · formula[i][j][k]
            if sum ≠ 0 then
                Break
        if sum ≠ 0 then                            ▷ If fails, go to next value
            Continue
```

Still inside of both while loops, we check the row condition for the current entry. If any of these fail, we go to the next iteration of the loop.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

for $0 \leq j' < j$ **do** ▷ Check row condition
  $sum \leftarrow 0$
  for $0 \leq k < n$ **do**
    $sum \leftarrow sum + formula[i][j'][k] \cdot formula[i][j][k]$
  if $sum \neq 0$ **then**
    Break
if $sum \neq 0$ **then** ▷ If fails, go to next value
  Continue

Inside of both while loops, we check the intercalacy condition for 2×2 submatrices. This means checking $v_{ij} \cdot v_{i'j'} + v_{i'j} \cdot v_{ij'} = 0$, letting v_{ij} denote the (i, j)th entry in $formula$.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

for $0 \leq i' < i,\ 0 \leq j' < j$ **do** ▷ Check intercalacy condition
  $sum \leftarrow 0$
  for $0 \leq k < n$ **do**
    $sum \leftarrow sum + formula[i'][j'][k] \cdot formula[i][j][k] + formula[i'][j][k] \cdot formula[i][j'][k]$
  if $sum \neq 0$ **then**
    Break
if $sum \neq 0$ **then** ▷ If fails, go to next value
  Continue

If we reach this point, all conditions have passed, so we have found an acceptable value for the current entry, and we end the inner while loop.

Once outside the loop, we can tell if we have found an acceptable value by checking sum. If we have not found an acceptable value, this means we have run out of values to try at the current entry. If we are at the first entry, this means there is no formula. Otherwise, we go

to the previous entry.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

```
        Break
    if number[i][j] ≥ ourSet.size() − 1 and sum ≠ 0 then
        number[i][j] ← −1
        if j = 0 and i = 1 then                ▷ If we are at the first entry,
            print "No formula found"           ▷ there is no formula
            Exit
        else if j = 0 then                     ▷ Otherwise, go back to previous entry
            i ← i − 1
            j ← r − 1
        else
            j ← j − 1
```

On the other hand, if we have found an acceptable value, we go to the next entry.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

```
    else
        if j < s − 1 then
            j ← j + 1
        else
            i ← i + 1
            j ← 0
```

We are now outside of the outer while loop, and have determined if a formula exists or not. If it exists, we print it. Otherwise, we print that no formula exists.

Algorithm 3 Existence of Sums-of-Squares Formulas over a Finite Field (cont'd)

```
print "Formula exists and is given by:"
for 0 ≤ i < r do                                   ▷ Print formula
    for 0 ≤ j < s do
        print "["
        for 0 ≤ k < n do
            print formula[i][j][k]
        print "]"
    print newline
```

For example, running this algorithm for $[6, 7, 8]$ produces the following matrix:

(10000000)	(01000000)	(00100000)	(00010000)	(00001000)	(00000100)
(01000000)	(20000000)	(00010000)	(00200000)	(00000100)	(00002000)
(00100000)	(00020000)	(20000000)	(01000000)	(00000010)	(00000001)
(00010000)	(00100000)	(02000000)	(20000000)	(00000001)	(00000020)
(00001000)	(00000200)	(00000020)	(00000002)	(20000000)	(01000000)
(00000100)	(00001000)	(00000002)	(00000010)	(02000000)	(20000000)
(00000010)	(00000001)	(00001000)	(00000200)	(00200000)	(00010000)

This corresponds to the following sums-of-squares formula (which is also a sums-of-squares formulas over the integers, when you replace 2 with -1):

$$z_1 = x_1y_1 + 2x_2y_2 + 2x_3y_3 + 2x_4y_4 + 2x_5y_5 + 2x_6y_6$$

$$z_2 = x_1y_2 + x_2y_1 + x_3y_4 + x_4y_3 + x_5y_6 + 2x_6y_5$$

$$z_3 = x_1y_3 + 2x_2y_4 + x_3y_1 + x_4y_2 + x_7y_5$$

$$z_4 = x_1y_4 + x_2y_3 + 2x_3y_2 + x_4y_1 + x_7y_6$$

$$z_5 = x_1y_5 + 2x_2y_6 + x_5y_1 + x_6y_2 + x_7y_3$$

$$z_6 = x_1y_6 + x_2y_5 + 2x + 5y_2 + x_6y_1 + x_7y_4$$

$$z_7 = x_3y_5 + 2x_4y_6 + 2x_5y_3 + x_7y_1$$

$$z_8 = x_3y_6 + x_4y_5 + 2x_5y_4 + 2x_6y_3 + x_7y_2$$

Note that because of the order in which we list the unit vectors, this algorithm will always produce formulas consisting of standard unit vectors first. This is useful for determining if there are sums-of-squares formulas over a finite field but not over the integers. If one would like to produce formulas involving non-standard unit vectors, one can change the start of the index $number[i][j]$ at various positions. This is done to produce a formula in the next section.

7.4 A New Formula

By implementing the algorithm from the previous section in characteristic 3 and adding the line

$$number[1][0] = 100$$

where indicated, we can produce a sums-of-squares formula of type $[3, 5, 7]$ over $\mathbb{F}_3$ which does not lift to a formula over the integers, although we will see that it is in the same orbit as an formula that lifts to the integers.

The algorithm from the previous section provides us with the output a matrix of vectors:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 2 & 0 & 2 & 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 2 & 1 & 0 & 0 & 2 & 2 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 2 & 2 & 1 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 2 & 0 & 0 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 2 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 2 & 1 & 0 & 2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 2 & 2 & 0 & 1 & 0 & 0 & 2 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 & 2 & 0 & 2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 & 1 & 0 & 0 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 0 & 0 & 2 & 2 & 0 \end{pmatrix}$$

This corresponds to the sums-of-squares formula of type $[3, 5, 7]$ over $\mathbb{F}_3$:

$$z_1 = x_1y_1 + 2x_2y_2 + x_3y_2 + x_4y_2 + 2x_5y_2 + 2x_2y_3 + x_3y_3 + 2x_4y_3 + x_5y_3$$
$$z_2 = x_2y_1 + 2x_3y_1 + 2x_4y_1 + x_5y_1 + x_1y_2 + x_2y_3 + x_3y_3 + 2x_4y_3 + 2x_5y_3$$
$$z_3 = x_2y_1 + 2x_3y_1 + x_4y_1 + 2x_5y_1 + 2x_2y_2 + 2x_3y_2 + x_4y_2 + x_5y_2 + x_1y_3$$
$$z_4 = x_2y_1 + x_3y_1 + x_2y_2 + x_4y_2 + x_3y_3 + x_4y_3$$
$$z_5 = 2x_4y_1 + 2x_5y_1 + x_2y_2 + x_4y_2 + 2x_2y_3 + 2x_5y_3$$
$$z_6 = x_2y_1 + x_3y_1 + 2x_3y_2 + 2x_5y_2 + 2x_2y_3 + 2x_5y_3$$
$$z_7 = x_4y_1 + x_5y_1 + x_3y_2 + x_5y_2 + 2x_3y_3 + 2x_4y_3$$

Although this formula cannot be lifted to the integers, it turns out that this formula is in the same orbit as a formula that can be lifted. Indeed, if we let

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 2 & 2 & 0 \\ 1 & 2 & 1 & 2 & 0 \\ 1 & 2 & 2 & 1 & 0 \end{pmatrix}^{-1} \in O(5, \mathbb{F}_3)$$

act on our formula, we obtain the formula:

$$\begin{pmatrix} 0 & 0 & 0 & 2 & 1 & 2 & 2 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 2 & 2 & 1 & 2 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 2 & 1 & 1 & 1 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 0 & 0 & 2 & 2 & 2 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 2 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 2 & 2 & 2 & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 2 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Note that, in this formula, for any pair of vectors, they are either orthogonal or the same (up to sign). This indicates that we can act on this to obtain a formula that lifts to the integers. Indeed, acting on this formula with the matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 1 & 2 & 1 & 2 \\ 0 & 0 & 0 & 2 & 1 & 1 & 2 \\ 0 & 0 & 0 & 2 & 2 & 1 & 1 \end{pmatrix}^{-1} \in O(7, \mathbb{F}_3)$$

we obtain the formula:

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 2 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

replacing 2's with -1's, we see that we obtain a formula over the integers.

Thus, although the formula does not lift to a formula over the integers, we have seen that it is in the same orbit as a formula that does lift to a formula over the integers.

By varying the choice 100 to other starting points and the starting points of other entries, we can find all sums-of-squares formulas of type $[3, 5, 7]$. This could potentially be used to determine the orbits of the action from Chapter 6.

7.5 Challenges and Constraints

The biggest challenge we face when conducting computer searches for sums-of-squares formulas is the sheer number of possibilities. Filling an $r \times s$ matrix with n colors with signs, there are $(2n)^{rs}$ possibilities. Filling an $r \times s$ matrix with n-dimensional vectors from a field with p elements, there are p^{rsn} possibilities. Even when we cut those numbers down by making clever choices, they still remain unwieldy.

For formulas over the integers, we would like to further reduce the number of possibilities with additional combinatorial arguments. However, we saw that the overhead introduced by organizing the possibilities actually resulted in the search being slower. Finding a more effective way to eliminate possibilities while minimizing the overhead is crucial to making these searches effective for larger values of n.

For formulas over finite fields, the most immediate challenge to making progress is that as n gets large, the number of unit vectors in $\mathbb{F}_p^n$ gets very large, very quickly. In addition, as the size of the field grows, the number of unit vectors explodes. Eventually this could become a memory issue, and, as it stands, just creating the list of unit vectors starts to take a very long time. Moving forward, we would like to emulate the methods of the refined search over the integers in order to eliminate many possibilities over finite fields. However, with such a long list of unit vectors, this can become unmanageable. Finding a way to effectively manage the long list of unit vectors should be a priority, and it may be necessary to find a completely new approach in order to make progress.

CHAPTER 8

Conclusion and Future Work

This dissertation has considered the questions:

- Is existence of a sums-of-squares formula independent of the base field?
- How can we effectively search for sums-of-squares formulas?

In considering these questions, we introduced a new approach to the study of sums-of-squares formulas. By introducing the variety of sums-of-squares formulas, we facilitate the application of a myriad of tools from algebraic geometry to sums-of-squares formulas. In this dissertation, we explored some of these applications, but many possibilities remain open.

We proved that, for algebraically closed fields, a sums-of-squares formula exists in characteristic 0 if and only if a formula exists in characteristic p for all but finitely many p. From this, we deduced that then a sums-of-squares formula will exist over some finite field, and we provided an upper bound for the degree of that field. These results were obtained using number theory and considering the zeta function. Thus we made progress on addressing the first question, showing partial independence of the existence of sums-of-squares formulas from the base field. Future work using number theory or the zeta function of this scheme may provide further gains on this problem.

We also used computational algebraic geometry to improve on our previous result: for algebraically closed fields, a sums-of-squares formula exists over a field of characteristic 0 if and only if a sums-of-squares formula exists over a field of characteristic p, for "large enough" p. This gave us an explicit bound on the "finitely many p" from Chapter 4. We achieved this by analyzing the coefficients that appear in Buchberger's algorithm. There is significant room

for improvement in our bound for "large enough," and studying the coefficients that appear throughout the algorithm could provide a definitive answer to the question of independence from the base field.

These results combine to show that existence of a sums-of-squares formula over an algebraically closed field is (theoretically, though not practically) computable.

We introduced an algebraic group action on the variety of sums-of-squares formulas, and we used this action to obtain a lower bound on the dimension of the variety of a sums-of-squares formulas, when it is nonempty. This action also provides us a new way to relate formulas to each other, which was utilized in writing algorithms to find formulas over finite fields. We still know very little about the structure of this action, and further study could provide new insights into the properties of the variety of sums-of-squares formulas.

We provided algorithms for finding sums-of-squares formulas over the integers and over finite fields. Conducting these computer searches is challenging, because as $[r, s, n]$ grow, the number of candidates for formulas becomes very large, very quickly. Further work to either improve these algorithms or find new algorithms for searching for sums-of-squares formulas could contribute significantly to the knowledge about sums-of-squares formulas. These searches could also provide a counterexample to our conjecture that existence of sums-of-squares formulas is independent of the base field, if this conjecture is false.

We still know very little about the structure of the variety of sums-of-squares formulas and our algebraic group action on this variety, in particular, we would like to know:

- When is the variety empty? Does it depend on the field F?
- Is the variety flat over Spec $\mathbb{Z}$? If it were flat, this would also answer the question of independence from the base field in the affirmative.
- Is the variety irreducible? Connected?
- Is the variety smooth? Non-singular? Regular?
- What is the variety's (exact) dimension?

- What do the components look like? How do they compare? Are there isolated points? What does the structure of the components mean for the sums-of-squares formulas?
- Is the group action faithful?
- What are the orbits? Is the action transitive? Is the variety of sums-of-squares formulas homogeneous?
- Is every sums-of-squares formula in the same orbit as a formula over the integers? This could show independence from the base field.
- What is the stabilizer of the action?
- What are some invariants of the group action? Can we find a useful invariant theory? This could help us conduct more efficient computer searches for formulas.

Answers to these questions would provide significant insight into the structure of the variety of sums-of-squares formulas, as well as potentially answering the question of whether existence of a sums-of-squares formula is independent of the base field.

References

[1] Adem, J. *On the Hurwitz Problem over an Arbitrary Field I,II.* Boletín de la Sociedad Matemática Mexicana, 25 (1980), 29-51; 26 (1981), 29-41.

[2] Atiyah, M.F. *Immersions and Embeddings of Manifolds.* Topology, 1 (1962), 125-132.

[3] Bombieri, E. *On Exponential Sums in Finite Fields, II.* Inventiones Mathematicae, 47 (1) (1978), 29-39.

[4] Clark, Pete L. *Quadratic Forms Chapter 1: Witt's Theory.* http://math.uga.edu/ pete/quadraticforms.pdf

[5] Dubé, Thomas W. *The Structure of Polynomial Ideals and Gröbner Bases.* SIAM Journal on Computing, 19 (4) (1990), 750-773.

[6] Dugger, Daniel and Isaksen, Daniel C. *The Hopf Condition for Bilinear Forms over Arbitrary Fields.* Annals of Mathematics, 165 (2007), 943-964.

[7] Dugger, Daniel and Isaksen, Daniel C. *Algebraic K-Theory and Sums-of-Squares Formulas.* Documenta Mathematica, 10 (2005), 357-366.

[8] Dugger, Daniel and Isaksen, Daniel C. *Etale Homotopy and Sums-of-Squares Formulas.* Mathematical Proceedings of the Cambridge Philosophical Society, 145 (2008), 1-25.

[9] Dwork, B. *On the Rationality of the Zeta Function of an Algebraic Variety.* American Journal of Mathematics, 82 (1960), 631-648.

[10] Eisenbud, David. *Commutative Algebra with a View Toward Algebraic Geometry.* Graduate Texts in Mathematics, 150 (1995).

[11] Grothendieck, A. *Formule de Lefschetz et Rationalité des Fonctions L.* Séminaire Bourbaki, 9 (1964-1966), 41-55.

[12] Hopf, H. *Ein Topologischer Beitrag zur Reellen Algebra.* Commentarii Mathematici Helvetici, 13 (1941), 219-239.

[13] Hurwitz, A. *Über die Komposition der Quadratischen Formen von Beliebig Vielen Variabeln.* Nachrichten von der Gesellschaft der Wissenshcaften zu Göttingen, Mathematisch-physikalische Klasse, (1898), 309-316.

[14] Hurwitz, A. *Über die Komposition der Quadratischen Formen.* Mathematische Annalen, 88 (1923), 1-25.

[15] Lam, K.Y. *Some New Results on Composition of Quadratic Forms.* Inventiones Mathematicae, 79 (1985), 467-474.

[16] Radon, J. *Lineare Scharen Orthogonaler Matrizen.* Abhandlungen Aus Dem Mathematischen Seminar Der Universitat Hamburg, 1 (1922), 1-14.

[17] Shapiro, Daniel B. *Compositions of Quadratic Forms.* Degruyter Expositions in Mathematics (2000).

[18] Shapiro, Daniel B. *Products of Sums of Squares, Lecture 1: Introduction and History.* https://people.math.osu.edu/shapiro.6/lec1.pdf

[19] Shapiro, Daniel B. *Products of Sums of Squares, Lecture 2: Integer Compositions.* https://people.math.osu.edu/shapiro.6/lec2.pdf

[20] Shapiro, Daniel B. *Products of Sums of Squares, Lecture 3: Arbitrary Fields.* https://people.math.osu.edu/shapiro.6/lec3.pdf

[21] Wan, Daqing. *Algorithmic Theory of Zeta Functions over Finite Fields.* MSRI Publications (44): Algorithmic Number Theory, 2008, 551-578.

[22] Weil, A. *Numbers of Solutions of Equations in Finite Fields.* Bulletin of the American Mathematical Society, 55 (1949), 497-508.

[23] Xie, Heng. *An Application of Hermitian K-Theory: Sums-of-Squares Formulas.* Documenta Mathematica, 19 (2014), 195-208.

[24] Yiu, P. *Sums of Squares Formulae with Integer Coefficients.* Canadian Mathematical Bulletin, 30 (1987), 318-324.

[25] Yiu, P. *On the Product of Two Sums of 16 Squares as a Sum of Squares of Integral Bilinear Forms.* Quarterly Journal of Mathematics: Oxford (2), 41 (1990), 463-500.

[26] Yiu, P. *Some Upper Bounds for Composition Numbers.* Boletín de la Sociedad Matemática Mexicana (3), 2(1996), 65-78.

[27] Yuzvinsky, Sergey. *Orthogonal Pairings of Euclidean Spaces.* Michigan Mathematical Journal, 28 (1981), 131-145.