UNIVERSITY OF CALIFORNIA,
IRVINE

On Calculating the Cardinality of the Value Set of a Polynomial (and some related problems)

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Mathematics

by

Joshua Erin Hill

Dissertation Committee:
Professor Daqing Wan, Chair
Associate Professor Vladimir Baranovsky
Professor Alice Silverberg

2014

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my appreciation to my thesis committee chair, Professor Daqing Wan, who helped me investigate a beautiful area of mathematics in my own way.

I would also like to thank my committee members, Professor Vladimir Baranovsky and Professor Alice Silverberg, for their wonderful courses, helpful comments, and interest in my research.

Finally, I would like to acknowledge the support and guidance of my wife, Laura Michelle Fulton.

Financial support was provided by the University of California, Irvine Mathematics department and NSF grants CCF0830701 and CCF1405564.

Joshua Erin Hill

Curriculum Vitæ

The machine does not isolate man from the great problems of nature but plunges him more deeply into them. – Saint-Exupéry

Education

2010 – 2014 **Doctor of Philosophy in Mathematics**, *University of California*, Irvine, December 2014.

2008 – 2010 **Masters of Science in Mathematics**, *University of California*, Irvine, December 10, 2010.

2005 – 2007 **Masters of Science in Mathematics**, *California Polytechnic State University*, San Luis Obispo, June 16, 2007.

1993 – 1999 **Bachelor of Science in Computer Science**, *California Polytechnic State University*, San Luis Obispo, December 11, 1999.

Doctoral Thesis

Title *On Calculating the Cardinality of the Value Set of a Polynomial (and some related problems)*

, Adviser Professor Daqing Wan

Teaching Training

- Two years of a weekly teaching seminar, Cal Poly, San Luis Obispo, 2005 — 2007.
- In-classroom evaluations and reviews by the Cal Poly, San Luis Obispo Mathematics Department.
- Two years of fortnightly teaching assistant seminars, UC Irvine, 2008 — 2012.
- In-classroom evaluation by the UC Irvine Teaching, Learning & Technology Center, April 2009.
- Advanced Teaching Assistant Seminar, Spring 2012.

josh-cv@untruth.org • *www.untruth.org*

Teaching Experience

2009–2014 **Graduate Student.**

Teaching Assistant at University of California at Irvine, Department of Mathematics. Concentration: Algebra / Algebraic Algorithmic Number Theory. Teaching assistant for 26 sections, each with 15 — 40 students, in the subjects of calculus (differential, integral, multi-dimensional), differential equations, linear algebra, introduction to abstract mathematics, group theory, ring theory, field theory, number theory, and cryptography. Co-instructor for a one-year, graduate-level algebra series. Organized and conducted the first mathematics department-initiated qualifying exam preparation problem-solving session to prepare graduate students for the algebra qualifying exam.

2005–2007 **Graduate Teaching Associate.**

Graduate Teaching Associate, for California Polytechnic State University. Instructor for 9 quarter-long university mathematics courses (Pre-calculus Algebra and Business Calculus). Developed syllabi, lectures, tests, quizzes, graded all student assignments and exams, and assigned final grades.

Awards

- *Outstanding Mathematics Teaching Assistant Award*, for 2010 — 2011
- *Graduate Student Mentor*, for 2012 — 2014.
- *Outstanding Contributions to the Department*, for 2013 — 2014.

Papers

1. *An Analysis of the RADIUS Authentication Protocol*, first published on the BUGTRAQ list in 2001, `http://untruth.org/s/p1.html`.
2. *Paul Erdős: Mathematical Genius, Human (in that order)*, `http://untruth.org/s/p2.html`.
3. *A Recurrence Relation for Pi*, `http://untruth.org/s/p4html`.
4. *Ciphertext stealing Wikipedia entry* (principal author), `http://en.wikipedia.org/wiki/Ciphertext_stealing`.
5. *An analysis of the "Guess 2/3 of the Average" game*, `http://untruth.org/s/p5.html`.
6. *A Description of the Number Field Sieve*, `http://untruth.org/s/p6.html`.
7. *The Minimum of n Independent Normal Distributions*, `http://untruth.org/s/p7.html`.
8. *The 7–11 Problem and its Solutions*, `http://untruth.org/s/p8.html`.
9. *Weil Image Sums (and some related problems)*, my 2011 advancement, `http://untruth.org/s/p9.html`.
10. *Counting Value Sets: Algorithm and Complexity* (with Qi Cheng and Daqing Wan), presented at the ANTS X conference 2012, `http://arxiv.org/pdf/1111.1224`.

josh-cv@untruth.org • *www.untruth.org*

Translated Papers

I translated (with help from Google Translate) Saburô Uchiyama's *Sur le Nombre des Valeurs Distinctes d'un Polynôme à Coefficients dans un Corps Fini*. My translation is *The Number of Distinct Values of a Polynomial with Coefficients in a Finite Field*, `http://untruth.org/s/p11.html`.

Public Presentations

- *The Zen of Information Security*, invited talk March 19th, 1999, for an undergraduate networking class.
- *Securing a Linux Box: It's mine, and You Can't Use It*, 2000, invited talk to the Cal Poly Linux Users Group.
- *Network Security: A Quick Overview*, 2002, invited talk to an undergraduate networking class, `http://untruth.org/s/p1002.html`
- *Cryptographic Foibles and Missteps*, 2008, invited talk to the Cuesta Computer Club, `http://untruth.org/s/p1003.html`
- *Coppersmith's Theorem: Background, Generalizations and Applications*, 2010 invited talk in the UCI Number Theory Seminar, `http://untruth.org/s/p1004.html`
- *Weil Image Sums (and some related problems)*, my 2011 advancement presentation, `http://untruth.org/s/p1005.html`
- *Counting Value Sets: Algorithm and Complexity*, my 2012 ANTS X presentation, `http://untruth.org/s/p1006.html`
- *Block Ciphers: Modes of Use, DES and AES*, 2012, a four hour-long presentations to a graduate cryptography class, `http://untruth.org/s/p1007.html`
- *Random Bit Generation: Theory and Practice*, 2013, an hour-long presentation to a graduate cryptography class, `http://untruth.org/s/p1008.html`
- *The Dual Elliptic Curve Deterministic RBG*, 2013, an hour-long presentation to a graduate cryptography class, `http://untruth.org/s/p1009.html`
- *Joux's Recent Index Calculus Results*, 2013, an invited two hour-long presentation to the UCI Number Theory Seminar, `http://untruth.org/s/p1010.html`
- *Substitution Ciphers*, an hour-long presentation to an undergraduate cryptography course, `http://untruth.org/s/p1011.html`
- *Harvey's Average Polynomial Time Algorithms*, 2014, an invited presentation to a UCI Arithmetic Geometry topics course, `http://untruth.org/s/p1012.html`
- *LaTeX for Mathy Endeavors: (Somewhat) Advanced LaTeX (and Related Matters)*, 2014, an invited presentation to the Anteater Mathematics Club, `http://untruth.org/s/p1013.html`
- *On Calculating the Cardinality of the Value Set of a Polynomial (and some related problems)*, 2014, my Thesis Defense presentation.

Internal Training Presentations

Each of these presentations was designed to run 2–8 hours.

- *Basic Cryptography*. Touches on historical uses of cryptography, the recent development of modern cryptography, cryptographic goals, cryptographic primitives, attack classes, security evaluation models, and a theoretical framework for symmetric and asymmetric cryptography.
- Cryptographic Algorithms. General principals of symmetric cipher design. Key schedules, general cipher design (Feistel and product ciphers). Detailed presentation of the design of DES, including weak/semi-weak keys and known attacks. Detailed presentation of the design of AES. Overview of internals of Skipjack, and SHA family.
- *Randomness Theory*. General theoretical background for RNG analysis and review, with emphasis on entropy evaluation of non-deterministic RNGs. Discussion on Shannon entropy and min-entropy. Summary of the SP800-22 testing requirements and use of the NIST sts tool.
- *Randomness Practice*. General PRNG design and characteristics. Detailed presentation on ANSI X9.31 A.2.4 PRNG, with emphasis on the algorithm's cycle properties. Implementation of the ANSI X9.31 A.2.4 PRNG using other symmetric algorithms. Detailed presentation on FIPS 186-2 appendix 3.1 PRNG, with emphasis on XSEED attacks. Detailed presentation on SP800-90 Hash_DRBG, HMAC_DRBG, CTR_DRBG. Summary of the findings for Dual_EC_DRBG.
- *Algorithm modes*. Discussion of symmetric algorithm confidentiality modes (ECB, CBC, CFB, OFB, CTR), including error propagation and plaintext malleability. Discussion of authentication modes (CBCMAC, CMAC, HMAC), including susceptibility to extension attacks. Discussion of combined modes (CCM, GCM)
- *Public/Private Key Cryptography*. Discussion of general properties of public/private systems, security strengths, and complete mathematical detail for RSA, DSA, ECDSA, DH, ECCDH, MQV and ECMQV. Demonstrate an example calculation for RSA, Diffie-Hellman, and ECDSA.
- *Error Detection Codes*. Basic error detection properties of parity, (ones' complement) checksum, and CRC. Examples of the calculation for each method.
- *Penetration Testing, The Path to Fun and Profit (through the inevitable)*. An overview of the techniques of penetration testing, with emphasis on the shortcomings of this testing technique.

Professional Experience

2010 2014 **Senior Security Analyst.**

Senior Security Analyst, on retainer for InfoGard Laboratories. Created white papers outlining current patterns in the information security industry. Consulting technical expert for the areas of network security evaluation, penetration testing, evaluation of random number generator entropy, wireless security standards, security standards production, e-prescribing security, NIST Computer Security Division and CMVP standards and interpretation, auditing statistical sampling, simple and differential power analysis and testing.

2008–2010 **Cryptography Consultant.**

Consultant, Technical consulting on hard problems. Performed standards interpretation and assessed design impacts on existing systems. Performed support for algorithm testing, including independent implementation of cryptographic algorithms to aid client testing. Assessed strength of cryptographic systems and protocols. Assisted in random number generator testing. Conducted design review of PED wireless protocols.

2004–2008 **Senior Security Engineer.**

Senior Security Engineer, for InfoGard Laboratories. In addition to the responsibilities of Security Engineer: Company technical lead. Provided technical guidance and training to security engineers and customers on complex technical issues. Evaluated formal models for high assurance systems. Performed design analysis and statistical evaluation of RNGs. Evaluated correctness and meaning of statistical tests. Authored, evaluated, and edited public ANSI/NIST security standards. Programmed and supported internal test tools. Performed simple and differential power analysis (SPA/DPA) and timing attack testing. Performed cryptographic protocol and algorithmic analysis. Developed FIPS 140-3 requirements and testing procedures. Participated in PCI scan vendor accreditation testing. Created InfoGard's Penetration Testing Laboratory, and was responsible for its operation.

1998–2004 **Security Engineer.**

Security Engineer, for InfoGard Laboratories. Performed FIPS 140-1 and 140-2 cryptographic module validation. Performed initial laboratory Common Criteria qualification test, and participated in Common Criteria testing of vendor products. Performed VISA PED and PCI testing. Performed USPS testing for electronic and mechanical indicia production. Conducted network security analysis. Produced written summaries of security vulnerabilities. Evaluated firewall and IDS designs and setup. Audited code as a portion of security evaluation. Performed system and network administration.

1997–1998 **Systems Developer.**

Systems Developer, for The Grid, a national ISP. Programmed and supported internal and external user interfaces. Supported DNS, mail, and web servers. Responsible for system and network security, including firewall design and implementation.

1996–1998 **System Administer.**

System Administrator, for Robert E. Kennedy Library, Cal Poly, San Luis Obispo. Initially set up and administered UNIX and Windows NT based computers. Installed and supported web, mail, DNS, gopher, and various custom network servers. Performed custom programming and scripting. Responsible for upkeep of legacy systems. Secured UNIX systems against threats, both internal and external.

Computer Languages

C, C++, Java, Perl, Bourne Shell, SQL, 80x86 Assembly, 680x0 Assembly, LaTeX, Mathematica, Z.

Abstract of the Dissertation

On Calculating the Cardinality of the Value Set of a Polynomial
(and some related problems)

By
Joshua Erin Hill
Doctor of Philosophy in Mathematics
University of California, Irvine, 2014
Professor Daqing Wan, Chair

We prove a combinatorial identity that relates the size of the value set of a map with the sizes of various iterated fiber products by this map. This identity is then used as the basis for several algorithms that calculate the size of the value set of a polynomial for a broad class of algebraic spaces, most generally an algorithm to calculate the size of the value set of a suitably well-behaved morphism between "nice" affine varieties defined over a finite field. In particular, these algorithms specialize to the case of calculating the size of the value set of a polynomial, viewed as a map between finite fields. These algorithms operate in deterministic polynomial time for fixed input polynomials (thus a fixed number of variables and polynomial degree), so long as the characteristic of the field grows suitably slowly as compared to the other parameters.

Each of these algorithms also produces a *fiber signature* for the map, which for each positive integer j, specifies how many points in the image have fibers of cardinality exactly j.

We adapt and analyze the zeta function calculation algorithms due to Lauder-Wan and Harvey, both as point counting algorithms and as algorithms for computation of one or many zeta functions.

These value set cardinality calculation algorithms extend to amortized cost algorithms that offer dramatic computational complexity advantages, when the computational cost is amortized over all the results produced. The last of these amortized algorithms partially answers a conjecture of Wan, as it operates in time that is polynomial in $\log q$ per value set cardinality calculated.

For the value set counting algorithms, these are the first such results, and offer a dramatic improvement over any previously known approach.

Chapter 0

Introduction

" 'When I use a word,' Humpty Dumpty said,
in rather a scornful tone, 'it means just what I
choose it to mean – neither more nor less.' "
Lewis Carroll, Through the Looking Glass

0.1 Setting

Study of algebraic maps often occurs via investigations of where those maps become zero (for suitably abstract notions of "zero"), and then some structural information about the map is drawn as a consequence. This approach has had a profound impact on the whole of mathematics, and has provided an invaluable and productive pattern of thought in all of algebra and its related disciplines.

Here we examine a related question: if we have a map f from the space X to the space Y, what can we say about the number of points in $f(X)$? More formally, denote

$$|V_f| = |\{f(x) : x \in X\}| .$$

We restrict ourselves to settings where $|V_f| < \infty$; sometimes this naturally occurs, and sometimes it requires restriction of the domain.

More generally, we examine the *fiber signature*, which is a specification of the number of points in the image that have exactly j elements in their respective fibers; the cardinality of the value set is then simply the sum of

the elements in the fiber signature. This more general information (at least in the category of sets) provides a large amount of information about a map between finite sets.

The primary example that we explore is that of affine varieties. It is instructive to note that in this setting, if we are given a suitably general algorithm for calculating the number of points in the value set of a morphism, we can also use this algorithm to count the number of points in the space by calculating the cardinality of the value set of the identity morphism $\mathrm{Id}_X : X \to X$. As such, the problem of counting the value set is in some sense a generalized version of the point counting problem.

Indeed, if you view these two problems from the setting of the polynomial hierarchy,[1] the point counting problem (where one counts the number of points in the domain that satisfy a set of constraints) is "lower" in the polynomial hierarchy than the problem of value set counting (where one counts points in the co-domain such that there exists a point in the domain satisfying a set of constraints).[2]

One result of this research has been to provide a connection in the other direction; we provide a way to use a point counting algorithm to solve the value set counting problem in the context of affine varieties. We thus see that in some sense, these problems are algorithmically closely related.

The size of the value set has been studied in various settings, but the most is known about the single variable polynomial case. In that setting, we examine a finite field with q elements, denoted $\mathbb{F}_q$ (with $q = p^a$, p prime), and take a positive degree polynomial $f \in \mathbb{F}_q[x]$ of degree d, and then examine $|V_f| = |f(\mathbb{F}_q)|$.

0.2 Computational Complexity

To compare approaches, we'll use the "big-oh" and "soft-oh" notations. Let A and B be two eventually positive real valued functions $A, B : \mathbb{N}^k \to \mathbb{R}$ under the norm $|\mathbf{x}|_{\min} = \min_i x_i$. The function A is said to be "big-oh" of B (written $A(\mathbf{x}) = O(B(\mathbf{x}))$) if and only if there exists a positive real constant C and an integer N so that if $|\mathbf{x}|_{\min} > N$ then $A(\mathbf{x}) \leq CB(\mathbf{x})$.

[1] Arora and Barak provide a nice introduction to the polynomial hierarchy.[2, Chp. 5]

[2] Notably, the value set counting problem involves an additional "there exists" quantifier!

Table 0.1: Notation

Notation	Description	Page
V_f	Value set of the function f.	1
$\mathbb{F}_q$	The finite field with q elements.	2
$O(\cdot)$	Big-Oh notation.	3
$\tilde{O}(\cdot)$	Soft-Oh notation.	3
$\lvert X\rvert$	Cardinality of the set X.	9
$\bar{\mathbb{F}}_q$	Some fixed algebraic closure of $\mathbb{F}_q$	15
$X(\mathbb{F}_{q^k})$	The $\mathbb{F}_{q^k}$-rational points on the variety X.	15
$X^{\times_Y k}$	The k-iterated fiber product of X.	37
$f\vert_{q^k}$	The function $f\vert_{X(\mathbb{F}_{q^k})}$ viewed as the function $f\vert_{q^k} : X(\mathbb{F}_{q^k}) \to Y(\mathbb{F}_{q^k})$.	53
$V_f\left(\mathbb{F}_{q^k}\right)$	The value set of f, viewed as a function on $\mathbb{F}_{q^k}$.	61
$\lceil\cdot\rceil$	Ceiling function.	
$\lfloor\cdot\rfloor$	Floor function.	
$\lVert f\rVert$	The maximum of the absolute values of the coefficients of the polynomial f.	
$\sigma_k(X_1, \cdots, X_m)$	The kth elementary symmetric polynomial on m variables.	
$\mathcal{O}(X)$	The ring of regular functions of the variety X.	

Similarly, A is said to be "soft-oh" of B (written $A(\mathbf{x}) = \tilde{O}(B(\mathbf{x}))$) if and only if there exists a positive real constant C' so that $A(\mathbf{x}) = O(B(\mathbf{x})\log^{C'}(B(\mathbf{x}) + 3))$. "Soft-oh" notation is used to dispense with log terms that might otherwise obscure the main thrust of "big-oh" notation.

0.3 Notation

The notation used within this paper is summarized in Table 0.1, and naming conventions surrounding polynomial and morphism degree are summarized in Table 0.2.

The general setting that we use in our principal findings (Section 0.4 is as follows: Let p be a prime, and a be a positive integer, with $q = p^a$. Let X and Y be algebraic varieties defined over $\mathbb{F}_q$.

Table 0.2: Naming Conventions

Notation	Description
d_i	Total degree of the ith polynomial.
d	One polynomial case: Total degree of the polynomial.
d	General affine case: Degree of a finite dominant morphism.
d_+	Sum of polynomial total degrees.
$\bar{d}$	Maximum of the polynomial total degrees.
$\mathcal{D}$	An upper bound on the number of $\mathbb{F}_{q^k}$-rational points in the fiber above any $\mathbb{F}_{q^k}$-rational point.

More precisely, Let X be an affine variety over $\bar{\mathbb{F}}_q$ defined by the vanishing set of (a non-negative integer) ℓ polynomials in affine r-space[3]

$$\alpha_1(x_1, \cdots, x_r) = \cdots = \alpha_\ell(x_1, \cdots, x_r) = 0,$$

where each $\alpha_i \in \mathbb{F}_q[x_1, \cdots, x_r]$.

Similarly, let Y be an affine variety over $\bar{\mathbb{F}}_q$ defined by the vanishing set of (a non-negative integer) m polynomials in affine s-space

$$\beta_1(y_1, \cdots, y_s) = \cdots = \beta_m(y_1, \cdots, y_s) = 0.$$

Denote the $\mathbb{F}_{q^w}$-rational points on X as $X(\mathbb{F}_{q^w})$, and additionally denote $x = (x_1, \cdots, x_r)$, and the analogous notions for y.

Let f be a morphism from X to Y which is an s-tuple of polynomials $f(x) = (f_1(x), \cdots, f_s(x))$, where each $f_i \in \mathbb{F}_q[x_1, \cdots, x_r]$.

For notational convenience, denote

$$d_i = \begin{cases} \deg \alpha_i & i \le \ell \\ \deg f_{i-\ell} & \ell < i \le \ell + s \\ d_{i \pmod{\ell+s}+1} & \text{otherwise} \end{cases},$$

and denote the restriction $f|_{X(\mathbb{F}_{q^k})}$ as $f|_{q^k}$, which is evidently a function $f|_{q^k} : X(\mathbb{F}_{q^k}) \to Y(\mathbb{F}_{q^k})$.

[3] As each of these polynomials provides a constraint, we operate under the convention that if $\ell = 0$, then $X = \mathbb{A}^r_{\bar{\mathbb{F}}_q}$, and similarly for the variety Y.

0.4 Principal Findings

Much of this paper turns on a pair of combinatorial findings[4] that apply to any map where the size of the map's fibers can be bounded (or through restriction can be made bounded). This first theorem relates the number of points in the k-iterated (set-wise) fiber products for positive integer k less than or equal to the size of the fiber bound to the number of points in the value set.

Theorem 14. *If X and Y are finite sets, and $f : X \to Y$ is a map such that any given fiber has at most d elements, then the cardinality of the image set of f is*

$$|V_f| = \sum_{i=1}^{d} (-1)^{i-1} N_i \sigma_i \left(1, \frac{1}{2}, \cdots, \frac{1}{d}\right),$$

where $N_k = \left|X^{\times_Y k}\right|$ and σ_i denotes the ith elementary symmetric function on d elements.

A similar argument also leads to the following finding, which can be repeated to calculate the full fiber signature.

Theorem 19. *If X and Y are finite sets, and $f : X \to Y$ is a map such that any given fiber has at most d elements, then for any positive integer $j \leq d$, the number of points in the co-domain whose fiber has exactly j elements is*

$$m_j = \binom{d}{j} \frac{1}{j} \sum_{i=1}^{d} (-1)^{i+j} N_i \sigma_{i-1} \left(1, \frac{1}{2}, \cdots, \frac{1}{j-1}, \frac{1}{j+1}, \cdots, \frac{1}{d}\right),$$

where $N_k = \left|X^{\times_Y k}\right|$ and σ_i denotes the ith elementary symmetric function on $d-1$ elements.

From this result (and suitable application of a point counting algorithm), we can calculate the image set of a morphism between two affine spaces, so long as the number of points in any fiber of the restricted morphism can be bounded.

[4] One of these combintorial findings was initially presented in a conference paper by Cheng-Hill-Wan.[10]

Theorem 23. *If there is a positive integer $\mathcal{D}$ so that $\left|\left(f|_q\right)^{-1}(y)\right| \leq \mathcal{D}$ for all $y \in Y(\mathbb{F}_q)$, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity*

$$\tilde{O}\left(2^{\mathcal{D}(\ell+s+r)-s}\mathcal{D}(\mathcal{D}r+2d_+\lambda+2\lambda)^{4\mathcal{D}r}\lambda^3 a^2 p^{1/2}\right) \text{ bit operations,}$$

where $\lambda = \max\left(a, \lceil(\mathcal{D}r+1)/2\rceil\right)$ and $d_+ = \sum_{i=1}^{\mathcal{D}\ell+(\mathcal{D}-1)s} d_i$.

This specializes to a number of important cases; first we discuss the polynomial case that has received the most attention.

Corollary 22. *Let a be a positive integer, p be a prime, $q = p^a$, and $f(x) \in \mathbb{F}_q[x]$ be a polynomial with positive degree d. There is a deterministic algorithm that calculates the cardinality of the value set, $|V_f|$ in $\mathbb{F}_q$, and more generally the fiber signature of f, with computational complexity*

$$\tilde{O}\left(2^{6d-1}\lambda^{4d+3}d^{8d+1}a^2p^{1/2}\right) \text{ bit operations,}$$

where $\lambda = \max\left(a, \lceil(d+1)/2\rceil\right)$.

We then proceed to deal with two more general situations that apply to the cardinality of the value set of finite morphisms on affine varieties. We first deal with the case where X is irreducible.

Corollary 25. *If X is irreducible and f is a finite dominant morphism from X to Y of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity described in Theorem 23, with $\mathcal{D} = d$.*

We then address the situation where we have some underlying information regarding the relation between the rings of regular functions associated with X and Y.

Corollary 26. *If f is a finite dominant morphism, and $\mathcal{O}(X)$ is generated by a set of t elements from $\mathcal{O}(Y)$ (via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism $f^\star$), then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity described in Theorem 23, with $\mathcal{D} = t$.*

One important special case of the above is

Corollary 27. *If f is a finite dominant morphism from $\mathbb{A}^r_{\overline{\mathbb{F}}_q}$ to $\mathbb{A}^r_{\overline{\mathbb{F}}_q}$ of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity*

$$\tilde{O}\left(2^{2dr-r} d(dr + 2d_+\lambda + 2\lambda)^{4dr} \lambda^3 a^2 p^{1/2}\right) \text{ bit operations,}$$

where $\lambda = \max\left(a, \lceil (dr+1)/2 \rceil\right)$ and $d_+ = \sum_{i=1}^{(d-1)r} d_i$.

We then move on to deal with the case where we perform larger calculations in hopes of getting better results after amortizing the cost per result. We refer to these as "amortized cost" algorithms.

We start with a method of calculating the cardinality of the value set of a morphism in many extensions of the base field. We do this by calculating several zeta functions, and then extracting the necessary information needed to calculate the size of the value set.

Theorem 28. *Let R be a positive integer. If there is a positive integer $\mathcal{D}$ so that $\left|(f|_{q^R})^{-1}(y)\right| \leq \mathcal{D}$ for all $y \in Y(\mathbb{F}_{q^R})$, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity*

$$\tilde{O}\Big(2^{\mathcal{D}(8\mathcal{D}r^2+17r+\ell+s)-s} \mathcal{D}^{4\mathcal{D}r+5} r^{4\mathcal{D}r+4} (d_+ + 2)^{\mathcal{D}r(4\mathcal{D}r+7)} a^{4\mathcal{D}r+4} p^{1/2} + R^2 a \mathcal{D}^2 r 2^{\mathcal{D}\ell+(\mathcal{D}-1)s} (4d_+ + 5)^{\mathcal{D}r} \log p\Big) \text{ bit operations,}$$

where $d_+ = \sum_{i=1}^{\mathcal{D}\ell+(\mathcal{D}-1)s} d_i$.

This finding can be specialized in the same way as our prior algorithm; we first discuss the results in the traditional single variable case.

Corollary 29. *Let a and R be positive integers, p be a prime, $q = p^a$, and f be a polynomial $f(x) \in \mathbb{F}_q[x]$, of positive degree d. There is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity*

$$\tilde{O}\left(2^{8d^2+18d-1} d^{8d^2+18d+5} a^{4d+4} p^{1/2} + R^2 2^{3d-1} d^{2d+2} a \log p\right) \text{ bit operations.}$$

We then consider the problem in the case of a finite dominant morphism of fixed degree from an irreducible variety.

Corollary 30. *If X is irreducible and f is a finite dominant morphism from X to Y of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity described in Theorem 28, with $\mathcal{D} = d$.*

A similar argument applies to the case where we have some underlying information regarding the relation between the ring of regular functions associated with X and Y.

Corollary 31. *If f is a finite dominant morphism, and $\mathcal{O}(X)$ is generated by a set of t elements from $\mathcal{O}(Y)$ (via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism $f^\star$), then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity described in Theorem 28, with $\mathcal{D} = t$.*

One important special case of the above (as in Corollary 27) can be found by letting $l = m = 0$, $s = r$, and $\mathcal{D} = d$ we arrive at

Corollary 32. *If f is a finite dominant morphism from $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ to $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity*

$$\tilde{O}\left(2^{d(8dr^2+18r)-r} d^{4dr+5} r^{4dr+4} \left(d_+ + 2\right)^{dr(4dr+7)} a^{4dr+4} p^{1/2} + R^2 a d^2 r 2^{(d-1)r} \left(4d_+ + 5\right)^{dr} \log p\right) \text{ bit operations,}$$

where $d_+ = \sum_{i=1}^{(d-1)r} d_i$.

We conclude by looking at the case where we perform this calculation for many primes at once in a somewhat less general setting, amortizing the cost of production of the zeta functions, as well as performing the value set counting calculation across many extensions at once.

Theorem 33. *Let r, s, N and R be positive integers. Let f be an s-tuple of polynomials $f(x) = (f_1(x), \cdots, f_s(x))$, where $f_i(x) = \mathbb{Z}[x_1, \cdots, x_r]$, where the total degree of f_i is d_i.*

If there is a positive integer $\mathcal{D}$ so that $\left|\left(f|_{p^R}\right)^{-1}(y)\right| \leq \mathcal{D}$ *for all* $y \in \mathbb{F}_{p^R}^s$ *and for all primes* $p < N$*, then there is a deterministic algorithm to calculate the cardinality of the value set of* $f|_{p^w}$*, and more generally the fiber signature of* $f|_{p^w}$*, for all* $w \leq R$ *and all primes* $p < N$*, with computational complexity*

$$\tilde{O}\left(2^{\mathcal{D}(8\mathcal{D}r^2+17r+s)-s+1}\mathcal{D}^{4\mathcal{D}r+8}r^{4\mathcal{D}r+6}((\mathcal{D}-1)d_+ + 2)^{\mathcal{D}r(4\mathcal{D}r+7)}N\log(\|f\|)+ N\mathcal{D}^2R^2r2^{(\mathcal{D}-1)s}\left(4(\mathcal{D}-1)d_+ + 5\right)^{\mathcal{D}r}\right) \text{ bit operations,}$$

where $d_+ = \sum_{i=1}^s d_i$ *and* $\|f\| = \prod_{j=1}^s \|f_j\|$.

This again can be applied to the single-variable case, where we get a dramatically better time complexity amortized per value-set result (as compared to calculating the size of the value set on a one-off basis.)

Corollary 34. *Let* N *and* R *be positive integers and* f *be a polynomial* $f(x) \in \mathbb{Z}[x]$ *of positive degree* d*. There is a deterministic algorithm to calculate the cardinality of the value set of the* p*-reduction* f *over* $\mathbb{F}_{p^w}$*, and more generally its fiber signature, for all positive integers* $w \leq R$ *and for all primes* $p \leq N$ *with computational complexity*

$$\tilde{O}\left(2^{d(8d+18)}d^{8d^2+18d+8}N\log(\|f\|) + NR^2 2^{3d-1}d^{2d+2}\right) \text{ bit operations.}$$

0.5 Prior Work Regarding the Value Set in One Variable

There are a few trivial bounds that can be immediately established; there are only q elements in the field, so $|V_f| \leq q$ (where $|\cdot|$ denotes the cardinality). Additionally, any polynomial of degree d can have at most d roots, thus for all $a \in V_f$, $f(x) = a$ is satisfied at most d times. This is true for every element in V_f, so $|V_f|\, d \geq q$, whence

$$\left\lceil \frac{q}{d} \right\rceil \leq |V_f| \leq q$$

(where $\lceil\cdot\rceil$ is the ceiling function).[5]

[5] This lower bound is commonly written $\lfloor (q-1)/d \rfloor + 1$, possibly in order to remain consistent with the notation used by Carlitz (*et al.*), who used this formulation in a setting where it was natural.[9]

Both of these bounds can be achieved: if $|V_f| = q$, then f is called a "permutation polynomial" and if $|V_f| = \lceil q/d \rceil$, then f is said to have a "minimal value set".

One way of exploring the behavior of $|V_f|$ is to look at asymptotic results that apply for many or most polynomials. Initial results by Uchiyama showed that if

$$f^*(u,v) = \frac{f(u) - f(v)}{u - v} \tag{1}$$

is absolutely irreducible, then $|V_f| > \frac{q}{2}$ for sufficiently large characteristic p.[49] He then showed that the requirement that (1) be absolutely irreducible could not be dropped.[8] In later work,[50] he established the average value for $|V_f|$ in terms of a value

$$\mu_d = 1 - \frac{1}{2!} + \frac{1}{3!} - \cdots + \frac{(-1)^{d-1}}{d!}.$$

This is just a power series expansion of $(1 - e^{-1})$, so as $d \to \infty$, μ_d quickly converges to this value. The *average* value across all polynomials was then seen to be

$$|V_f| \sim \mu_d q + O(1).$$

Birch and Swinnerton-Dyer made this estimate more concrete for a class of polynomials that they somewhat optimistically called "general polynomials"[4] (that is those polynomials such that the Galois group of $f(x) - t$ over $\bar{\mathbb{F}}_q(t)$ is the symmetric group on d elements). So long as f is a general polynomial, we have $\mu = \mu_d$, and

$$|V_f| = \mu q + O_d\left(q^{1/2}\right).$$

They also proved that μ depends only on d and two Galois groups:

$$G(f) = \mathrm{Gal}\left(f(x) - t/\mathbb{F}_q(t)\right)$$
$$G^+(f) = \mathrm{Gal}\left(f(x) - t/\bar{\mathbb{F}}_q(t)\right)$$

Cohen refined this and provided an explicit statement for μ in terms of Galois groups.[12] Let K be the splitting field for $f(x) - t$ over $\mathbb{F}_q(t)$ and $k' = K \cap \bar{\mathbb{F}}_q$. Finally, define:

$$G^*(f) = \{\sigma \in G(f) : K_\sigma \cap k' = \mathbb{F}_q\}$$
$$G_1(f) = \{\sigma \in G(f) : \sigma \text{ fixes at least one point}\}$$
$$G_1^*(f) = G_1(f) \cap G^*(f)$$

Cohen found that we then have $\mu = |G_1^*| / |G^*|$.[6]

Voloch showed [51] that for general q, the Galois group condition described by Birch and Swinnerton-Dyer [4] implies that the surface $f^*(x, y) = 0$ meets the smoothness requirement $\partial^2 y/\partial x^2 \neq 0$, which he demonstrated was sufficient to provide a lower bound on $|V_f|$:

$$|V_f| \geq \frac{2q^2}{(d+1)q + (d-1)(d-2)}.$$

The problem of establishing $|V_f|$ has been studied in various forms for at least the last 115 years, but exact closed form expressions for $|V_f|$ are known only for polynomials in very specific forms. The behavior of $|V_f|$ when f is constant or degree 1 is clear ($|V_f| = 1$ and $|V_f| = q$, respectively). Kantor partially solved the cubic case (mod 3)[29], and then Uchiyama completely characterized $|V_f|$ for f of degree 2 ($p \neq 2$) or 3 ($p \neq 2, 3$).[49]

For higher degree polynomials, exact formulae for $|V_f|$ are only known for polynomials in a few special forms. The special case of the p-linear polynomial is fairly straight forward: for linear operators, the size of the image is just the ratio of the total size of the space divided by the kernel of the map.

Dickson Polynomials of the first kind have been well studied, and their image set is completely understood (this class includes the cyclic polynomial X^d.[11]). Cusick determines the exact value for $|V_f|$ for $f(X) = X^k(1 + X)^{2^m-1}$ in $\mathbb{F}_{2^{2m}}$, for $k = \pm 1, \pm 2,$ or 4 [13] and for $f(X) = (X+1)^d + X^d + 1$ for particular values of d over $\mathbb{F}_{2^m}$.[14]

More is known about polynomials that fall into the special cases that we have already introduced: permutation polynomials (including exceptional polynomials) and polynomials with minimal value sets. There are few permutation polynomials known (indeed, permutation polynomials are asymptotically fairly sparse. A randomly selected polynomial is a permutation polynomial with probability e^{-q} for large q. [19])

Dickson classified all permutation polynomials of degree less than or equal to six in his thesis.[17] Additional classes of permutation polynomials include certain parameter sections of Dickson Polynomials of the first and second kind, reversed Dickson Polynomials, Linearized Polynomials, and polynomials of the form $x^{(q+1)/2} + ax$.[7]

[6] This provides a wonderful combinatorial explanation of Uchiyama's μ_d, as the proportion of non-derangements in S_d.

[7] Lidl and Niederreiter provide a wonderful introduction on this topic.[35, chp. 7]

Hayes moved the question of characterizing permutation polynomials into the realm of algebraic geometry by noting that f is a permutation polynomial if and only if the variety defined over $\bar{\mathbb{F}}_q^2$ by $f^*(X,Y)$ only has $\mathbb{F}_q$-rational points on the diagonal $X = Y$.[28] This approach became the study of exceptional polynomials, those polynomials such that the factorization of $f^*(X,Y)$ into irreducibles in $\mathbb{F}_q[X,Y]$ contains no absolutely irreducible terms (that is, each irreducible term in the factorization must *not* be irreducible in $\bar{\mathbb{F}}_q[X,Y]$). The characteristic of being an exceptional polynomial was recognized quite early as being very closely related to that of being a permutation polynomial. Cohen proved that almost all exceptional polynomials were also permutation polynomials,[12] and Wan removed the last special cases.[52] A consequence of the Lang-Weil bound is that if $p \nmid d$, $d > 1$ and $q > d^4$, then any permutation polynomial of degree d is also an exceptional polynomial. Thus, for sufficiently large fields, the notions of permutation polynomial and exceptional polynomial are largely the same.

There have been a few notable algorithms to test to see if a polynomial is a permutation polynomial; this is relevant, because detecting if a polynomial is a permutation polynomial or a polynomial with a minimal value set is a specialization of the value set counting problem, so these algorithms can be seen as closely related algorithms to algorithms that count the size of the value set. This characteristic of polynomials was used by Ma and von zur Gathen to provide a ZPP (Zero-error Probabilistic Polynomial time) algorithm for testing a polynomial to determine if it is a permutation polynomial.[36] Shparlinski provided a fully deterministic test that determines if a given polynomial is a permutation polynomial by extending prior work due to von zur Gathen [20] to an algorithm that has time complexity $\tilde{O}((dq)^{6/7})$ for all d and q.[45] More recently, in 2005, Kayal made a deterministic polynomial time algorithm that tests to see if a polynomial is a permutation polynomial.[30]

There are numerous results that provide bounding inequalities for $|V_f|$, average values for $|V_f|$ (summed across all polynomials up to degree $q-1$) and asymptotic results for $|V_f|$, but these largely do not lead to exact values for $|V_f|$. One notable exception is Wan's proof of Mullen's conjectured bound for non-permutation polynomials:[41][52]

$$|V_f| \leq \left\lfloor q - \frac{q-1}{d} \right\rfloor.$$

This bound was found to be sharp by Cusick and Müller ($f(X) =$

$(X+1)\,X^{q-1}$ achieves this bound).[15] Thus, if any polynomial is found to have more distinct points in the image than allowed by this bound, then it must be a permutation polynomial.

A similar finding by Gomez-Calderon showed that if a low degree polynomial has a sufficiently small value set, then it must have a minimal value set.[23] In particular, if f is a polynomial of degree $3 \leq d < p$ and

$$|V_f| \leq \left\lfloor \frac{q-1}{d} \right\rfloor + 2\left(\frac{q-1}{d^2}\right) - 1$$

then f has a minimal value set.

These two findings act to form "exclusion zones": certain disallowed values for $|V_f|$ for polynomials of particular degrees.[8]

Several families of polynomials with minimal value sets have been discovered. All polynomials with minimal value sets with degree $d < 2p+2$ were classified by Carlitz, Lewis, Mills, and Straus, [9] and then Mills continued by further classifying all polynomials of degree $d \leq \sqrt{q}$.[39]

Significant additional work in this area was performed by Javier Gomez-Calderon in his doctoral thesis and then later in collaboration with Madden. In these papers, he characterizes all polynomials of degree $d < \sqrt[4]{q}$ for which $|V_f| < 2q/d$; many of these polynomials result in forms based on Dickson polynomials.[22][24]

As we have seen, the exact value for $|V_f|$ is known in only very limited cases.

0.6 Prior Work on the Multi-Dimensional Case

Clearly, any result in the multi-dimensional case must be weaker than the corresponding single dimensional case, so there isn't a great deal to add.

Sun explored a restricted value set of a multi-variable polynomial over a finite field, where each variable is chosen from a subset of the finite field; this proved particularly nice when the size of the subset of the full space being considered had the right form.[48]

[8] Given these "exclusion zone" restrictions, one might get the false impression that polynomials must take only certain types of images; this is incorrect! To dispel this notion, note that one can construct a polynomial that takes any arbitrary value set by using Lagrange Interpolation.

There has been some recent progress at showing results similar to the "exclusion zone" style findings of the single-variable case. Mullen, Wan and Wang found that if a polynomial in n variables isn't a permutation polynomial, then the cardinality of its value set is

$$|V_f| \leq q^n - \min\left\{\frac{n(q-1)}{d}, q\right\}.[42]$$

One interesting approach to the multi-variable case is reducing it to the single variable case. The polynomial $f \in (\mathbb{F}_q[x_1, \cdots, x_n])^n$ can be regarded as a single map on $\mathbb{F}_{q^n}$, by simply noting that as a vector space, if $e_1, \cdots, e_n$ are an $\mathbb{F}_q$-basis for $\mathbb{F}_{q^n}$, then

$$\mathbb{F}_{q^n} \cong \bigoplus_{i=1}^{n} \mathbb{F}_q e_i,$$

so we can naturally consider the points as being written under this coordinate system, so $x = (x_1, \cdots, x_n) = x_1e_1 + \cdots + x_ne_n$ and similarly $f = (f_1(x), \cdots, f_n(x)) = f_1e_1 + \cdots + f_ne_n$. As any map over a finite field (in this case $\mathbb{F}_{q^n}$) has the same behavior as a polynomial map[9], we can thus pass our multi-variable f to a single variable $\tilde{f} : \mathbb{F}_{q^n} \to \mathbb{F}_{q^n}$. Doing this amusing shuffle reduces the multi-variable case to a single variable, but this approach yields a somewhat Pyrrhic victory from the perspective of trying to understand the algebraic complexity of the map f. The resulting degree is expected to be quite large (on the order of q^n), so we lose the information provided by the degree.

Kosters provided an approach relying on the "q-degree" of f, a quantity derived from the sum of the digits of the base-q expansion of each of the degrees. This approach yields an "exclusion zone" analogous to that seen in the single variable case, namely that if f is not a permutation polynomial, then

$$|V_f| \leq q^n - \frac{n(q-1)}{d},$$

where $d = \max_i \deg f_i$.[32]

[9]The explicit polynomial can be calculated by using Lagrange interpolation.

Chapter 1

Algorithms Involving The Weil Zeta Function

> "You look at where you're going and where you are and it never makes sense, but then you look back at where you've been and a pattern seems to emerge. And if you project forward from that pattern, then sometimes you can come up with something."
>
> *Robert M. Pirsig*, Zen and the Art of Motorcycle Maintenance

Examine the affine variety described by the simultaneous zeros of polynomials, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ over some fixed algebraic closure of $\mathbb{F}_q$, denoted $\overline{\mathbb{F}}_q$; call this variety X.[10]

Each finite extension of $\mathbb{F}_q$ is isomorphic to a field of the form $\mathbb{F}_{q^k}$ for some k; denote the set of $\mathbb{F}_{q^k}$-rational points on the variety X (the set of simultaneous zeros within the extension $\mathbb{F}_{q^k}$) as $X(\mathbb{F}_{q^k})$, that is

$$X(\mathbb{F}_{q^k}) = \left\{ x \in \mathbb{F}_{q^k}^n : f_1(x) = \cdots = f_m(x) = 0 \right\}.$$

[10] Some authors use the term "affine variety" to mean "irreducible algebraic set", but we do not adopt this convention. The affine varieties here are merely algebraic sets unless they are explicitly described as being irreducible.

We can now define the zeta function for our set of polynomials:

$$Z_X = Z_X(T) = \exp\left(\sum_{k=1}^{\infty} \frac{\left|X(\mathbb{F}_{q^k})\right|}{k} T^k\right). \tag{1.1}$$

The zeta function clearly contains a profound amount of information about the polynomial set. Our immediate task is that of counting the number of $\mathbb{F}_q$-rational points on a variety defined over $\bar{\mathbb{F}}_q$, that is we are looking for the number of simultaneous zeros of our polynomial set, where each coordinate lies in $\mathbb{F}_q$. Thus, if we can calculate the zeta function, we should be able to extract the number of $\mathbb{F}_q$-rational points of the variety X.[11]

From the definitions this seems like a less-than-useful statement, but surprises abound in mathematics! Weil conjectured that the zeta function is a rational function; this was first proven by Dwork using p-adic methods, and then later proven by ℓ-adic cohomological methods by Grothendieck. [18][26] The common zeros of our polynomial set are not expected to form any particularly nice variety (non-singular projective, a curve, an abelian variety, etc.) so there are very few options for efficiently performing point counting or calculating $Z_X(T)$ explicitly.

In this paper, we make use of a few related algorithms that were originally specified in order to explicitly calculate $Z_X(T)$. These algorithms calculate $Z_X(T)$ through use of a *trace function* which is used to extract $\left|X(\mathbb{F}_{q^k})\right|$ for some necessary number of terms. In the instance where we require the zeta function, the corresponding papers provide complexity results that we can adapt to our setting. We also use these algorithms in order to directly calculate $|X(\mathbb{F}_q)|$, which requires some additional development.

We first describe how we efficiently adapt these more general algorithms to this task; for each algorithm we provide an analysis of the time required to calculate $|X(\mathbb{F}_q)|$ for a variety defined by a single polynomial, and then describe how to adapt these results to more general settings.[12]

We then move on to the general case of full zeta functions, where we present the full algorithms, adapt them to our setting, and then describe how to extend them to the general affine case.

[11] See Section 1.2 for an algorithm that accomplishes this on a grand scale.

[12] For an introduction to this area, the expository papers of Wan and Lauder nicely outline an approach which is fundamentally enabled by Dwork.[53][33][18]

1.1 Point Counting

1.1.1 Lauder and Wan's Point Counting Algorithm

For a polynomial of total degree d in n variables, Lauder and Wan described an algorithm that explicitly calculates the zeta function of any variety defined by the zeros of one polynomial which runs in polynomial time so long as the characteristic grows suitably slowly compared to the other terms (on the order of $p = O((d \log q)^C)$ for some positive constant C).[53][34] This algorithm is based on a toric point counting algorithm, which can be adapted to count points on more general spaces.

Proposition 1 (Lauder and Wan)**.** *Let a and n be positive integers, p be a prime, $q = p^a$, and $f \in \mathbb{F}_q[x_1, \cdots, x_n]$ be a polynomial of positive degree d. There is a deterministic algorithm that calculates the number of solutions of $f(x_1, \cdots, x_n) = 0$ residing in $\mathbb{F}_q^n$ in*

$$\tilde{O}\left(2^n a^{3n+7} n^{3n+5} d^{3n} p^{2n+4}\right) \text{ bit operations.}$$

Proof. Lauder and Wan accomplish this by iterated use of a toric point counting algorithm [34, Algorithm 29] that can be used to piece together the total number of points in $\left|X(\mathbb{F}_{q^k})\right|$. More explicitly, if we start with a polynomial $f(x_1, \cdots, x_n) \in \mathbb{F}_q[x_1, \cdots, x_n]$ of total degree d, then this algorithm calculates

$$N^* = \left|\left\{(x_1, \cdots, x_n) \in \left(\mathbb{F}_q^*\right)^n : f(x_1, \cdots, x_n) = 0\right\}\right|$$

in $\tilde{O}\left(a^{3n+7} n^{3n+5} d^{3n} p^{2n+4}\right)$ bit operations[13].[34, Proposition 36] This is not exactly what we want (this is the number of points in the affine torus!), but we can find the corresponding $X(\mathbb{F}_q)$ by examining the affine torus decomposition of $\mathbb{F}_q^n$, and then summing.[14]

We classify all the points in our space by the location of zeros in their affine coordinates. Denote the set of coordinate indices that are identically

[13] In fact, if we know the shape of the polytope corresponding to f, then Lauder and Wan's algorithm can do still better than this!

[14] This is a restricted version of the approach described in Lauder-Wan, where they assemble the full zeta function in this way.[34]

0 as $S \subset \{1, 2, \cdots, n\}$, and denote the corresponding torus as

$$T_S^n = \left\{(x_1, \cdots, x_n) \in \mathbb{F}_q^n : x_i = 0 \text{ if and only if } i \in S\right\}.$$

We then count the number of zeros in the related space

$$\left|X\left(\mathbb{F}_q\right)^S\right| = \left|\{x \in T_S^n : f(x) = 0\}\right|.$$

The zeros counted for each of the 2^n distinct selections of S are clearly disjoint, and any given zero is in one of the resulting sets, so we can simply sum and calculate

$$|X(\mathbb{F}_q)| = \sum_{S \subset \{1, \cdots, n\}} \left|X\left(\mathbb{F}_q\right)^S\right|.$$

Each $\left|X\left(\mathbb{F}_q\right)^S\right| \leq q^{n-|S|}$, so their sum is less than

$$\begin{aligned} B &= \sum_{S \subset \{1, \cdots, n\}} q^{n-|S|} \\ &= \sum_{i=1}^{n} \binom{n}{i} q^{n-i} \\ &< (q-1)^n. \end{aligned}$$

The additions can thus take place in $O(2^n n a \log p)$ bit operations, which is dominated by the cost of the point counting.

This yields a total computational complexity of $\tilde{O}\left(2^n a^{3n+7} n^{3n+5} d^{3n} p^{2n+4}\right)$ bit operations to calculate $|X(\mathbb{F}_q)|$. □

1.1.2 Harvey's Point Counting Algorithm

Proposition 2 (Harvey)**.** *Let a and n be positive integers, p be a prime, $q = p^a$, $\lambda = \max(a, \lceil (n+1)/2 \rceil)$, and $f \in \mathbb{F}_q[x_1, \cdots, x_n]$ be a polynomial of positive degree d. There is a deterministic algorithm that calculates the number of solutions of $f(x_1, \cdots, x_n) = 0$ residing in $\mathbb{F}_q^n$ in*

$$\tilde{O}\left(2^n (n + 2d\lambda + 2\lambda)^{4n} \lambda^3 a^2 p^{1/2}\right) \text{ bit operations.}$$

Proof. In Harvey's algorithm, we examine the space $\mathbb{P}^n_{\bar{\mathbb{F}}_q}$, projective n-space over $\mathbb{F}_q$, with homogeneous coordinates $x_0, \cdots, x_n$. If we then have a homogeneous polynomial ${}^h f \in \mathbb{F}_q[x_0, \cdots, x_n]$ of total degree d such that $p \nmid d$, Harvey's algorithm [27, Theorems 1.2-1.3] allows us to explicitly calculate the zeta function of the projective variety cut out of the affine torus by ${}^h f$.

In this projective n-space, we count the number of (projective) points within the affine torus satisfying the equation, that is

$$\begin{aligned} N^*_{\text{proj}} &= \left| \left\{ [x_0 : \cdots : x_n] \in \mathbb{P}^n_{\bar{\mathbb{F}}_q} : {}^h f(x_0 : \cdots : x_n) = 0 \text{ and } \prod_{i=0}^{n} x_i \neq 0 \right\} \right| \\ &= \frac{1}{q-1} \left| \left\{ (x_0, \cdots, x_n) \in \left(\mathbb{F}^*_q\right)^{n+1} : {}^h f(x_0, ..., x_n) = 0 \right\} \right|. \end{aligned}$$

The computational complexity of calculating this value requires some abuse of the framework developed in Harvey's paper; we start with Harvey's simplified trace formula [27, Theorem 3.1]

$$N^*_{\text{proj}} \equiv (q-1)^n \sum_{s=0}^{\lambda} (-1)^s \binom{\lambda}{s} \operatorname{tr}\left(A^a_{F^s}\right) \pmod{p^\lambda},$$

and set λ large enough so that the resulting value is equality and simultaneously so that $\lambda \geq (n+1)/2$. To accomplish this, we set $\lambda = \max(a, \lceil (n+1)/2 \rceil)$.

By a lemma on computing the trace function,[27, Lemma 3.4] given a set of companion matrices, M_s, we can evaluate the $\operatorname{tr}\left(A^a_{F^s}\right)$ in

$$(n + 2d\lambda)^{3n} \lambda \log^{1+\epsilon}(2\lambda) a^{2+\epsilon} \log^{2+\epsilon} p \text{ bit operations.}$$

We can use the deformation-based technique developed by Harvey,[27, Proposition 4.4] which gives us the ability to calculate each M_s in

$$(n + 2d\lambda)^{4n} \lambda^2 \log^{1+\epsilon}(2\lambda) a^{1+\epsilon} p^{1/2} \log^{2+\epsilon} p \text{ bit operations.}$$

Putting these together, in order to calculate N^*_{proj}, we see that we must calculate $(\lambda + 1)$ distinct M_s values and traces, which occurs in

$$O\left((n + 2d\lambda)^{4n} \lambda^3 \log^{1+\epsilon}(2\lambda) a^{2+\epsilon} p^{1/2} \log^{2+\epsilon} p\right) \text{ bit operations.}$$

Denote the set of indices that are identically 0 as $S \subset \{0, 1, 2, \cdots, n\}$, and denote the corresponding affine torus within $\mathbb{P}^n_{\bar{\mathbb{F}}_q}$ as

$$\hat{T}^n_S = \left\{ [x_0 : \cdots : x_n] \in \mathbb{P}^n_{\bar{\mathbb{F}}_q} : \text{ and } x_i = 0 \text{ if and only if } i \in S \right\}.$$

We then count the number of zeros in the related space

$$\left| X^{\text{proj}} \left(\mathbb{F}_q \right)^S \right| = \left| \left\{ x \in \hat{T}^n_S : {}^h f(x) = 0 \right\} \right|.$$

Summing over every permissable choice of S would give us the full $|X^{\text{proj}}(\mathbb{F}_q)|$, but we are interested in the points in a related space.

Instead, we want the number of affine points of a (possibly) non-homogeneous polynomial. If we start with an arbitrary polynomial $f(x_1, \cdots, x_n) \in \mathbb{F}_q[x_1, \cdots, x_n]$ of degree $d > 0$, with $p \nmid d$, then we can apply homogenization and arrive at

$${}^h f(x_0, \cdots, x_n) = x_0^d f\left(\frac{x_1}{x_0}, \cdots, \frac{x_n}{x_0} \right),$$

a homogeneous polynomial of total degree d. If $p \mid d$, then we can modify this by instead letting

$${}^h f(x_0, \cdots, x_n) = x_0^{d+1} f\left(\frac{x_1}{x_0}, \cdots, \frac{x_n}{x_0} \right),$$

which is a homogeneous polynomial of total degree $d + 1$, but now p does not divide the degree of ${}^h f$.

In either case, the affine roots of f correspond to the projective roots where $x_0 \neq 0$, as these are equivalent to projective zeros of ${}^h f(1, x_1, \cdots, x_n)$. We further see that each such distinct projective zero corresponds to exactly one affine zero of $f(x_1, \cdots, x_n)$, so the value we seek is actually calculated as

$$|X(\mathbb{F}_q)| = \sum_{S \subset \{1, \cdots, n\}} \left| X^{\text{proj}} \left(\mathbb{F}_q \right)^S \right|.$$

Each $\left| X^{\text{proj}} \left(\mathbb{F}_q \right)^S \right| \leq q^{n-|S|}$ so, just as with the prior bound, their sum

$$B < (q-1)^n.$$

The additions can thus take place in $O(2^n n a \log p)$ bit operations, which is dominated by the cost of the point counting.

This yields a total computational complexity of at worst

$$\tilde{O}\left(2^n(n+2(d+1)\lambda)^{4n}\lambda^3 a^2 p^{1/2}\right) \text{ bit operations}$$

to calculate $|X(\mathbb{F}_q)|$. □

1.1.3 Point Counting on Affine Varieties

There are several techniques that can be used to extend the algorithms discussed in sections 1.1.1 and 1.1.2 to the multi-polynomial setting that we started with. In most cases, the second approach we outline is asymptotically faster.

As before, we examine the variety described by the simultaneous zeros of polynomials, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ over the field $\bar{\mathbb{F}}_q$; call this variety X. We fix the notation $x = (x_1, \cdots, x_n)$ and $d_i = \deg f_i(x)$.

1.1.3.1 Reduction to a Single Hypersurface

This approach reduces the case of many polynomials to that of a single polynomial, along with a counting-based technique to move between these two situations. This approach is similar to a tool used in calculation of zeta functions due to Gao.[53, §3]

Corollary 3. *Let a, n, and m be positive integers, p be a prime, $q = p^a$, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ be polynomials of positive degree, where each f_i has total degree d_i, and $\bar{d} = \max_i d_i$. There is a deterministic algorithm that calculates the number of simultaneous solutions of $f_1(x_1, \cdots, x_n) = \cdots = f_m(x_1, \cdots, x_n) = 0$ residing in $\mathbb{F}_q^n$ in*

$$\tilde{O}\left(2^{n+m}a^{3(n+m)+7}(n+m)^{3(n+m)+5}(\bar{d}+1)^{3(n+m)}p^{2(n+m)+4}\right) \text{ bit operations.}$$

Corollary 4. *Let a, n, and m be positive integers, p be a prime, $q = p^a$, $\lambda = \max(a, \lceil (n+m+1)/2 \rceil)$, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ be polynomials of positive degree, where each f_i has total degree d_i, and $\bar{d} = \max_i d_i$. There is a deterministic algorithm that calculates the number of simultaneous solutions of $f_1(x_1, \cdots, x_n) = \cdots = f_m(x_1, \cdots, x_n) = 0$ residing in $\mathbb{F}_q^n$ in*

$$\tilde{O}\left(2^{n+m}(n+m+2\bar{d}\lambda+4\lambda)^{4(n+m)}\lambda^3 a^2 p^{1/2}\right) \text{ bit operations.}$$

Proof. Examine the set $\{x \in \mathbb{F}_q^n : f_1(x) = \cdots = f_m(x) = 0\}$. We can represent these m distinct polynomial constraints within as a single polynomial over $\mathbb{F}_q$ by introducing additional variables $z_1, \cdots, z_m$. Denote $z = (z_1, \cdots, z_m)$, and examine the function

$$F(x, z) = \sum_{i=1}^{m} z_i f_i(x). \tag{1.2}$$

Let X denote the affine variety defined by the simultaneous zeros of $f_1, \cdots, f_m$ over $\bar{\mathbb{F}}_q$. If $\gamma \in X(\mathbb{F}_q)$, then $F(\gamma, z)$ is the zero function. We note that we have added a total of m extra variables, so for each such choice of $\gamma \in X(\mathbb{F}_q)$ there are q^m distinct zeros of F. On the other hand, if $\gamma \in \mathbb{F}_q^n \setminus X(\mathbb{F}_q)$, then the solutions of $F(\gamma, z) = 0$ specify a $(m-1)$-dimensional linear subspace of $\mathbb{F}_q^m$, so for any such γ there are q^{m-1} zeros of F.

These two cases are clearly disjoint, so if we denote the cardinality of the solution set to $F(x, z) = 0$ as $|F|$, then we see that

$$\begin{aligned} |F| &= q^m |X(\mathbb{F}_q)| + q^{m-1} (q^n - |X(\mathbb{F}_q)|) \\ &= |X(\mathbb{F}_q)| q^{m-1} (q-1) + q^{n+m-1}. \end{aligned}$$

Solving for $|X(\mathbb{F}_q)|$, we find that

$$|X(\mathbb{F}_q)| = \frac{|F| - q^{n+m-1}}{q^{m-1}(q-1)} \tag{1.3}$$

thus we have an easy way of solving for $|X(\mathbb{F}_q)|$ given $|F|$.

The total degree of $F(x, z)$ is $1 + \max d_i$, and the polynomial is now in $n + m$ variables. Plugging these values into propositions 1 and 2 yield the desired result. □

1.1.3.2 Application of the Principle of Inclusion / Exclusion

Here, a combinatorial approach is used to count points in the intersection of zeros of each function; this approach is related to an approach to calculating the full zeta functions described by Wan.[15]

[15] This approach is outlined in Section 1.3.3.

Corollary 5. *Let a, n, and m be positive integers, p be a prime, $q = p^a$, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ be polynomials of positive degree, where each f_i has total degree d_i, and $d_+ = \sum_i d_i$. There is a deterministic algorithm that calculates the number of simultaneous solutions of $f_1(x_1, \cdots, x_n) = \cdots = f_m(x_1, \cdots, x_n) = 0$ residing in $\mathbb{F}_q^n$ in*

$$\tilde{O}\left(2^{n+m} a^{3n+7} n^{3n+5} {d_+}^{3n} p^{2n+4}\right) \text{ bit operations.}$$

Corollary 6. *Let a, n, and m be positive integers, p be a prime, $q = p^a$, $\lambda = \max(a, \lceil (n+1)/2 \rceil)$, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ be polynomials of positive degree, where each f_i has total degree d_i, and $d_+ = \sum_i d_i$. There is a deterministic algorithm that calculates the number of simultaneous solutions of $f_1(x_1, \cdots, x_n) = \cdots = f_m(x_1, \cdots, x_n) = 0$ residing in $\mathbb{F}_q^n$ in*

$$\tilde{O}\left(2^{n+m} (n + 2d_+\lambda + 2\lambda)^{4n} \lambda^3 a^2 p^{1/2}\right) \text{ bit operations.}$$

Proof. The principle of inclusion / exclusion [7, p.76] allows us to find the number of points in some universal set that are not present in any of a finite list of subsets. We'll dwell on this statement somewhat in order to make later calculations more natural; if we have m sets $A_1, \cdots, A_m$ in some universal set A, and we denote $A_j^c = A \setminus A_j$, then one standard formulation of the principle of inclusion / exclusion is

$$\left| \bigcap_{i=1}^{m} A_i^c \right| = \sum_{I \subset \{1, \cdots, m\}} (-1)^{|I|} \left| \bigcap_{i \in I} A_i \right| .$$

If we let $B_i = A_i^c$, then this gives us

$$\begin{aligned}
\left|\bigcap_{i=1}^{m} B_i\right| &= \sum_{I\subset\{1,\cdots,m\}} (-1)^{|I|} \left|\bigcap_{i\in I} B_i^c\right| \\
&= \sum_{I\subset\{1,\cdots,m\}} (-1)^{|I|} \left(|A| - \left|\bigcup_{i\in I} B_i\right|\right) \\
&= \underbrace{\sum_{I\subset\{1,\cdots,m\}} (-1)^{|I|} |A|}_{0} - \sum_{I\subset\{1,\cdots,m\}} (-1)^{|I|} \left|\bigcup_{i\in I} B_i\right| \\
&= \sum_{I\subset\{1,\cdots,m\}} (-1)^{|I|-1} \left|\bigcup_{i\in I} B_i\right|
\end{aligned}$$

We lastly note that the empty union is the empty set, so this gives us the formulation

$$\left|\bigcap_{i=1}^{m} B_i\right| = \sum_{\emptyset\neq I\subset\{1,\cdots,m\}} (-1)^{|I|-1} \left|\bigcup_{i\in I} B_i\right|, \tag{1.4}$$

which is the version we want to use.

We'll use X to denote the variety over $\bar{\mathbb{F}}_q$ defined by the simultaneous zeros of all the f_i polynomials, and for any $I \subset \{1, \cdots, m\}$ write

$$f_I(x) = \prod_{i\in I} f_i(x),$$

and finally use X_I to denote the corresponding variety over $\bar{\mathbb{F}}_q$ defined by the zeros of f_I.

We can evidently calculate $|X_I(\mathbb{F}_q)|$ using one of our point counting algorithms, where f_I is a polynomial in n variables of degree $d_I = \sum_{i\in I} d_i$. We then have the following statement in terms of q-rational points on the varieties we've described:

$$|X(\mathbb{F}_q)| = \sum_{\emptyset\neq I\subset\{1,\cdots,m\}} (-1)^{|I|-1} |X_I(\mathbb{F}_q)|. \tag{1.5}$$

This requires a total of $2^m - 1$ invocations of one of the point counting algorithms. The largest computation necessary is associated with $I = \{1, \cdots, m\}$; in this case computing $|X_I(\mathbb{F}_q)|$ requires counting the number of zeros of f_I, a polynomial in n variables of degree

$$d_+ = \sum_{i=1}^{m} d_i.$$

We can bound the cost of this approach by counting $2^m - 1$ invocations of a point counting algorithm, each acting on a polynomial in n variables of degree d_+.

Note that we have a trivial bound for the size of our terms, namely $0 \leq |X_I(\mathbb{F}_q)| \leq q^n$ and for that matter, $0 \leq |X(\mathbb{F}_q)| \leq q^n$. Even if we ignore the alternating nature of this sum, we get the trivial bound for our sum of $q^n 2^m$ and such numbers can be added in $O(nam \log p)$ bit operations. This occurs $(2^m - 1)$ times, so the addition occurs in time complexity $\tilde{O}(2^m na \log p)$ bit operations, which is dominated by the point counting operation. □

1.2 Point Counting From the Zeta Function

Thus far, we've analyzed counting points and ignored the possibility of using a zeta function to extract the number of $\mathbb{F}_{q^r}$-rational points. Here we instead consider the case where we take as input the full zeta function $Z_X(T)$ for some variety over $\mathbb{F}_q$, and then extract the number of points $|X(\mathbb{F}_{q^r})|$ for all r up to some positive integer bound, R.

All of the algorithms for calculating the zeta function that we have discussed start by calculating $|X(\mathbb{F}_{q^r})|$ for all positive r up to some bound greater than the degree of either the numerator or denominator, so we can consider as having started with these values, and in particular we can assume that R is greater than the degree of the numerator or denominator of the zeta function.

We start by recalling that the zeta function is of the form

$$\begin{aligned} Z_X(T) &= \exp\left(\sum_{r \geq 1} \frac{|X(\mathbb{F}_{q^r})|}{r} T^r\right) \\ &= \frac{g(T)}{h(T)}, \end{aligned}$$

where $g, h \in 1+T\mathbb{Z}[T]$. Taking the logarithmic derivative of this expression yields

$$\sum_{r\geq 1} \left|X\left(\mathbb{F}_{q^r}\right)\right| T^{k-1} = \frac{g'(T)}{g(T)} - \frac{h'(T)}{h(T)}. \tag{1.6}$$

As such, if we are given the zeta function as a rational function, Equation (1.6) reduces the problem of counting the number of points in $X(\mathbb{F}_{q^r})$ to the problem of finding the $(r-1)$th term of the formal power series of these two rational expressions, for which we use the following lemma.

Lemma 7. *If $g \in 1 + T\mathbb{Z}[T]$, then the first R terms of the formal power series $g'(T)/g(T)$ can be deterministically calculated in $\tilde{O}(R^2 \log \|g\|)$ bit operations, where $\|g\|$ denotes the maximum of the absolute values of the coefficients of g.*

Proof. The algorithms presented here perform optimally when R is a power of 2, but there are more complicated versions that do not require this and which have essentially the same computational complexity, so we can without loss of generality assume that $R = 2^t$ for some positive integer t.

This task can be accomplished by computing some truncation of the formal power series for these rational functions. We'll continue with the notation $g(T) = 1 + \sum_{i=1}^{d_g} g_i T^i$ (a unit of the ring of formal power series), then to find the first 2^t terms of the multiplicative inverse of this series.

For the purpose of a bit-operation-oriented analysis, we use Kronecker substitution for multiplication.[21, §8.4] We seek to multiply the polynomials $h_1, h_2 \in 1 + T\mathbb{Z}[T]$. We can without loss of generality assume the coefficients of these polynomials are uniformly positive; if not we can break each of the polynomials into a difference of two polynomials with uniformly positive coefficients, and then combine by using at most four iterations of the same algorithm. This results in no change in the time complexity (in big-oh or soft-oh notation).

Denote $d_i = \deg h_i$, the maximum coefficient of $h_i(T)$ in absolute value as $\|h_i(T)\|$, and $\ell_i = \lfloor \log_2 \|h_i(T)\| \rfloor + 1$. We let $d = \max(d_1, d_2)$, and $\ell = \max(\ell_1, \ell_2)$, the number of bits required to store any coefficient in either polynomial. We then "evaluate" our polynomials at 2^ℓ ("evaluation" here is just a matter of shifting parameters into the appropriate place in the binary representation of an integer, so the evaluation occurs in $O(d\ell)$ bit operations). We can assume that $d < 2^t$, as we know all the coefficients up to the degree, so we can thus multiply the polynomials h_1 and h_2 by

multiplying the resulting integers, and read out the results in $\tilde{O}(2^t \ell)$ bit operations.

For finding the inverse, we use Sieveking-Kung[16] to calculate the power series inverse. For ease of reference, the variant of Sieveking-Kung that we analyze is specified in Algorithm 1.

Algorithm 1: Truncated Polynomial Inverse

input : $g \in 1 + T\mathbb{Z}[T]$ and $t \in \mathbb{N}$
output: $h \in 1 + T\mathbb{Z}[T]$ such that $gh \equiv 1 \pmod{x^{2^t}}$

$g_0 \leftarrow 1$ $i \leftarrow 1$ **while** $i \leq t$ **do**
  $g_i \leftarrow 2g_{i-1} - fg_{i-1}^2 \pmod{x^{2^t}}$ $i \leftarrow i + 1$
end while
return g_t

To analyze the computational complexity of this algorithm, we first specify a recurrence relation in terms of $\ell_g = \lfloor \log \|g\| \rfloor + 1$ to calculate the maximal bit length of any coefficient in g_i, which we'll denote as l_i. For ease of representation as a generating function[17], we adopt the convention that $l_k = 0$ for $k < 0$, and note that $l_0 = 1$, $l_1 = \ell_g$, $l_2 = 3\ell_g$, and generically

$$l_k = 2l_{k-1} + \ell_g - 2\delta(k-1) - (\ell_g - 1)\delta(k) \text{ for } k \geq 0,$$

where $\delta(\cdot)$ denotes the Dirac delta function.

The generating function associated with these l_i values is then

$$\mathscr{L}(z) = \sum_{i \geq 0} l_i z^i;$$

substituting this into the recurrence relation gives us

$$\mathscr{L}(z) = 2z\mathscr{L}(z) + \frac{\ell_g}{1-z} - 2z - (\ell_g - 1),$$

which, when converted into series notation and simplified, gives us

$$\mathscr{L}(z) = \sum_{i \geq 1} \ell_g \left(2^i - 1\right) z^i + 1.$$

[16] A nice treatment of this approach is discussed by von zur Gathen.[21, §9.1]

[17] Graham, Knuth and Patashnik provide a lovely introduction to these matters.[25, Chapter 7]

This gives us a closed form, namely

$$l_i = \ell_g \left(2^i - 1\right) \text{ for } j \geq 1.$$

Each g_i polynomial has degree less than[18] 2^t. Together, we find that the entire inversion operation occurs in $\tilde{O}(2^{2t}\ell_g)$ bit operations and yields a polynomial of degree $2^t - 1$ whose maximal coefficient is length no larger than $\ell_g(2^t - 1)$.

We then multiply by $g'(T)$, which has no term larger than $\ell_g + \lfloor \lg d_g \rfloor + 1$ and is degree $d_g - 1$, resulting in a total computational complexity of $\tilde{O}(2^{2t}\ell_g)$ bit operations. □

From this, along with some bounds on the coefficients and degree of $g(T)$ and $h(T)$, we can extract $X(\mathbb{F}_{q^r})$ from the zeta function.

1.3 Algorithms for Calculating the Zeta Function

We will find cause to calculate the full zeta function for spaces, so for completeness, we state two results that we will use, and then extend these to a slightly more general setting.

1.3.1 Calculating Single Zeta Functions

The following theorem is directly due to Lauder and Wan.[34, Theorem 37]

Theorem 8 (Lauder and Wan)**.** *Let a and n be positive integers, p be a prime, $q = p^a$, and X be a variety defined over $\bar{\mathbb{F}}_q$ defined by the vanishing set of $f \in \mathbb{F}_q[x_1, \cdots, x_n]$, where f is of total degree d. There is a deterministic algorithm that calculates the zeta function of X in*

$$\tilde{O}\left(2^{13n^2} a^{3n+7} d^{3n^2+9n} p^{2n+4}\right) \text{ bit operations.}$$

The corresponding theorem in Harvey must again be adapted for our use.

[18] Indeed, by using a similar approach to the above, we could find that the degree of g_i is equal to $\min(2^t - 1, (2^i - 1)d_g)$, but we won't need this.

Theorem 9 (Harvey)**.** *Let a and n be positive integers, p be a prime, $q = p^a$, and X be a variety defined over $\bar{\mathbb{F}}_q$ defined by the vanishing set of $f \in \mathbb{F}_q[x_1, \cdots, x_n]$, where f is of total degree d. There is a deterministic algorithm that calculates the zeta function of X in*

$$\tilde{O}\left(2^{8n^2+17n} n^{4n+4} (d+2)^{4n^2+7n} a^{4n+4} p^{1/2}\right) \text{ bit operations.}$$

Proof. This is a consequence of a theorem by Harvey.[27, Theorem 1.3] As previously mentioned, we examine the space $\mathbb{P}^n_{\mathbb{F}_q}$, projective n-space over $\mathbb{F}_q$, with homogeneous coordinates $x_0, \cdots, x_n$. If we then have a homogeneous polynomial ${}^h f \in \mathbb{F}_q[x_0, \cdots, x_n]$ of total degree d such that $p \nmid d$, Harvey's algorithm [27, Theorems 1.2-1.3] allows us to explicitly calculate the zeta function of the projective variety cut out by ${}^h f$ from the affine torus. In the event that $p \mid d$, then we simply replace ${}^h f$ with $x_0 {}^h f$, which is now degree $d+1$ (which p does not divide) and which has the same zeta function on the affine torus.[19] In this (worst) case, we then can calculate this zeta function in

$$2^{8n^2+16n} n^{4n+4+\epsilon} (d+2)^{4n^2+7n+\epsilon} a^{4n+4+\epsilon} p^{1/2} \log^{2+\epsilon} p \text{ bit operations.}$$

Our specification of the Weil zeta function is fundamentally described in terms of point counting; multiplying zeta functions corresponds directly to adding the number of points in each finite extension of $\mathbb{F}_q$ (and similarly dividing corresponds to subtracting points). As such, if we can represent any space as a union of disjoint subvarieties, we can then calculate the zeta function of the full variety by just taking the product of the zeta functions of the subvarieties. We can thus directly apply the same techniques (and notation) that we used in the proof of Proposition 2 to adapt Harvey's algorithm to our use.

We again denote ${}^h f$ as the homogenization of the polynomial f, and denote $Z^S_{X^{\text{proj}}}(T)$ as the zeta function associated with the vanishing set of ${}^h f$ on $\hat{T}^n_S$. We now can find $Z_X(T)$, through multiplying

$$Z_X(T) = \prod_{S \subset \{1, \cdots, n\}} Z^S_{X^{\text{proj}}}(T).$$

[19] In the affine torus, $x_0 \neq 0$ so $x_0 {}^h f(x) = 0$ if and only if ${}^h f(x) = 0$. This is true in all extensions, so the zeta function must be the same.

The resulting zeta function is a rational function, say $g(T)/h(T)$, where $g, h \in 1 + T\mathbb{Z}[T]$. Bombieri proved, for a variety defined by one polynomial in n variables of degree d (in reduced form) that $\deg(g(T) + h(T)) < (4d + 5)^n$.[5, Theorem 1] We'll call this bound D, and we'll use this as a bound for the bound of the degree of either g or h.

We now seek a bound on the coefficients of the zeta function. First, recall that the zeta function is of the form

$$Z_X(T) = \exp\left(\sum_{k=1}^{\infty} \frac{\left|X(\mathbb{F}_{q^k})\right|}{k} T^k\right).$$

In the complex plane, the exponential function is entire and has no zeros, so any value of T that causes the exponent to converge to a finite complex value could not correspond to a zero or a pole. As such, the zeros and poles evident in the factorization of the zeta function's numerator and denominator must cause the series in the exponent to diverge. Examining the power series in the exponent of the zeta function

$$\sum_{k \geq 1} \frac{\left|X(\mathbb{F}_{q^k})\right|}{k} T^k,$$

we see if $|T| < q^{-n}$, then the series will certainly converge. Writing the zeta function as a ratio of products of linear terms gives us

$$g(T) = \prod_i (1 - \beta_i T) \qquad\qquad h(T) = \prod_j (1 - \gamma_j T),$$

where the β_i are the reciprocal zeros of $g(T)$ and the γ_j are the reciprocal zeros of $h(T)$. As se see from the above, this series diverges for both the zeros and poles of the zeta function, so this gives us

$$\frac{1}{\beta_i} \geq q^{-n} \qquad\qquad \frac{1}{\gamma_j} \geq q^{-n}$$
$$\beta_i \leq q^n \qquad\qquad \gamma_i \leq q^n.$$

We thus find that any coefficient of either g or h (which is a product of some number of these reciprocal zeros) is thus bounded by q^{Dn}.

Using this coefficient bound, we then see that in reduced form

$$B = \log \max \{\|g(T)\|, \|h(T)\|\} \leq \log \left(q^{Dn}\right),$$

so we can bound the computational complexity of the multiplication as $\tilde{O}(DB)$ bit operations. Putting the resulting zeta function back into reduced form requires a polynomial multiplication and polynomial GCD calculation, at cost $\tilde{O}(DB)$ bit operations, and then two polynomial divisions, also at cost $\tilde{O}(DB)$ bit operations. Thus, we see that the total computational complexity associated with multiplication and reduction of the zeta function is

$$\begin{aligned}\tilde{O}(2^n DB) &= \tilde{O}(2^n D^2 na \log p) \\ &= \tilde{O}(2^n (4d+5)^{2n} a \log p) \text{ bit operations,}\end{aligned}$$

which is dominated by the cost of computing the zeta functions.

Together, this yields a total computational complexity of

$$\tilde{O}\left(2^{8n^2+17n} n^{4n+4} (d+2)^{4n^2+7n} a^{4n+4} p^{1/2}\right) \text{ bit operations.}$$

to calculate $Z_X(T)$. □

1.3.2 Calculation of a Family of Zeta Functions

If we are given a polynomial $f(x_1, \cdots, x_n) \in \mathbb{Z}[x_1, \cdots, x_n]$ of total degree d, and a prime p, we can reduce the coefficients of this polynomial mod p (resulting in the p-reduction of f), and then examine the variety formed by the zero set of the p-reduction of $f(x_1, \cdots, x_n)$ over $\bar{\mathbb{F}}_p$. Harvey noticed that we can apply a memoization-style technique to this calculation when we calculate all the zeta functions for such varieties associated with all primes less than some bound N.

Theorem 10 (Harvey)**.** *Let n and N be positive integers, $f \in \mathbb{Z}[x_1, \cdots, x_n]$ be a polynomial of total degree d, and denote the maximum coefficient of f in absolute value as $\|f\|$. For a prime p let X_p denote the affine variety defined over $\bar{\mathbb{F}}_p$ defined by the vanishing set of the p-reduction of f. There is a deterministic algorithm to calculate the zeta function of X_p for all $p < N$ in*

$$\tilde{O}\left(2^{8n^2+17n+1} n^{4n+6} (d+2)^{4n^2+7n} N \log \|f\|\right) \text{ bit operations.}$$

Proof. This is a consequence of another theorem of Harvey.[27, Theorem 1.4] Harvey's algorithm calculates a family of zeta functions, one for each prime $p \nmid d$ less than N. Each variety is the space the (p-reduced) homogeneous polynomial, say ${}^h f$, cuts out of the affine torus in projective n-space in

$$2^{8n^2+16n} n^{4n+6+\epsilon} (d+1)^{4n^2+7n+\epsilon} N \log^2(N) \log^{1+\epsilon}(N\|f\|) \text{ bit operations.}$$

We would like to have the complete list, not just these, so we also examine the zeta functions associated with $x_0 {}^h f$, which as we've seen before has the same zeta function (as we are within the affine torus, so $x_0 \neq 0$). Between these two, we have the complete set of zeta functions for X_p for all $p < N$. This complete calculation thus occurs in

$$2^{8n^2+16n+1} n^{4n+6+\epsilon} (d+2)^{4n^2+7n+\epsilon} N \log^2(N) \log^{1+\epsilon}(N\|f\|) \text{ bit operations.}$$

As in Theorem 9, we must perform this calculation a total of 2^n times and then multiply together the corresponding results. There are less than N zeta functions total.

We note that the degree bound from the proof of Theorem 9 is the same as here, that is $D < (4d+5)^n$. For a coefficient bound, note that any prime in our list is less than N, so our coefficient bound gives us $B < \log\left(N^{Dn}\right)$, so we can again bound the computational complexity of the multiplication and re-reduction operations as $\tilde{O}(DB)$ bit operations. Thus, we see that the total computational complexity associated with all the multiplications and reductions of the zeta functions is

$$\begin{aligned} \tilde{O}(N2^n DB) &= \tilde{O}(N2^n D^2 n \log N) \\ &= \tilde{O}(N2^n (4d+5)^{2n} \log N) \text{ bit operations,} \end{aligned}$$

which is dominated by the cost of computing the zeta functions.

This yields a total computational complexity of

$$\tilde{O}\left(2^{8n^2+17n+1} n^{4n+6} (d+2)^{4n^2+7n} N \log \|f\|\right) \text{ bit operations}$$

to calculate $Z_{X_p}(T)$ for all $p < N$. □

Consider the case where the polynomial being examined is fixed. The above algorithm then has computational complexity

$$O\left(N \log^{3+\epsilon}(N)\right) \text{ bit operations,}$$

that is to say that we have a quasilinear time algorithm in N.

If N is large, then the number of primes less than or equal to N is asymptotically

$$\pi(N) \sim \frac{N}{\log N}.$$

Dividing by the total number of primes, the amortized cost per prime of the above algorithm thus has time complexity $O\left(\log^{4+\epsilon}(N)\right)$ bit operations, that is to say the cost per produced zeta function is polylogarithmic time for such a fixed polynomial.

1.3.3 Extension to Affine Varieties

Adopting the notation and approach from corollaries 5 and 6, we now examine the variety described by the simultaneous zeros of polynomials, $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$; call this variety X. We fix the notation $x = (x_1, \cdots, x_n)$ and $d_i = \deg f_i(x)$.

By combining the above observation that multiplication of zeta functions translates to adding the number of points in each finite extension of $\mathbb{F}_q$ (and division to subtracting points) with the principle of inclusion/exclusion described in Section 1.1.3 we can extract full zeta function.[20] This approach directly yields the following corollaries.

Corollary 11. *Let a, n, and m be positive integers, p be a prime, $q = p^a$, X be a variety over $\bar{\mathbb{F}}_q$ defined by the simultaneous vanishing set of the polynomials $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ with positive total degrees d_i, and $d_+ = \sum_i d_i$. There is a deterministic algorithm that calculates the zeta function of X in*

$$\tilde{O}\left(2^{13n^2+m} a^{3n+7} {d_+}^{3n^2+9n} p^{2n+4}\right) \text{ bit operations.}$$

Corollary 12. *Let a, n, and m be positive integers, p be a prime, $q = p^a$, $\lambda = \max\left(a, \lceil (n+1)/2 \rceil\right)$, X be a variety over $\bar{\mathbb{F}}_q$ defined by the simultaneous vanishing set of the polynomials $f_1, \cdots, f_m \in \mathbb{F}_q[x_1, \cdots, x_n]$ with positive*

[20] This approach is essentially one of those described by Wan.[53, §3]

total degrees d_i. Denote $d_+ = \sum_i d_i$. There is a deterministic algorithm that calculates the zeta function of X in

$$\tilde{O}\left(2^{8n^2+17n+m} n^{4n+4} (d_+ + 2)^{4n^2+7n} a^{4n+4} p^{1/2}\right) \text{ bit operations.}$$

Proof. We continue where we left off in Section 1.1.3.2. Note that our Equation (1.5) applies not just to the base field, but every finite extension of that field, that is:

$$|X(\mathbb{F}_{q^r})| = \sum_{\emptyset \neq I \subset \{1,\cdots,m\}} (-1)^{|I|-1} |X_I(\mathbb{F}_{q^r})|. \tag{1.7}$$

Recalling that products of zeta functions correspond to adding the count of points within the corresponding extensions, and division of zeta functions correspond to subtracting the count of points within the corresponding extensions, we find that we can in some sense calculate Equation (1.7) for all values of r by multiplying and dividing by the corresponding zeta functions. This yields the pleasant result

$$Z_X(T) = \prod_{\emptyset \neq I \subset \{1,\cdots,m\}} Z_{X_I}(T)^{(-1)^{|I|-1}}.$$

Calculating $Z_X(T)$ then requires a total of $(2^m - 1)$ invocations of Theorems 8 or 9, the most costly of which is associated with the variety $X_{\{1,\cdots,m\}}$.

In order to bound the difficulty of performing this calculation, we first try to bound the degree of the resulting zeta function. The resulting zeta function is a rational function, say $g(T)/h(T)$. The m-polynomial version of the Bombieri bound yields a degree bound

$$\begin{aligned} D &\leq \sum_{\emptyset \neq I \subset \{1,\cdots,m\}} (4d_I + 5)^n \\ &\leq \sum_{i=1}^{m} \binom{m}{i} (4d_+ + 5)^n \\ &< 2^m (4d_+ + 5)^n. \end{aligned}$$

We again apply our coefficient bound and find that $B = Dan \log p$.

Each multiplication, and the following reduction, can be done in $\tilde{O}(DB)$ bit operations. There are a total of 2^m of these, so that means all the multiplication required for the principle of inclusion/exclusion can be accomplished in

$$\begin{aligned}
&\tilde{O}\left(2^m D^2 an \log p\right) = \\
&\tilde{O}\left(2^{3m} \left(4d_+ + 5\right)^{2n} an \log p\right) = \\
&\tilde{O}\left(2^{3m} \left(4d_+ + 5\right)^{2n} a \log p\right) \text{ bit operations.}
\end{aligned} \tag{1.8}$$

This is dominated by the zeta function calculations. □

We again examine the situation discussed in Section 1.3.2, that is we now consider our polynomials as having integer coefficients. Once the coefficients of these polynomials are reduced, we have the same situation as above.

Corollary 13. *Let n, m, and N be positive integers, $f_1, \cdots, f_m \in \mathbb{Z}[x_1, \cdots, x_n]$ be polynomials with positive total degrees d_i and maximal coefficients $\|f_i\|$, with $\|f\| = \prod_i \|f_i\|$. For a prime p let X_p denote the affine variety defined over $\bar{\mathbb{F}}_p$ defined by the simultaneous vanishing set of all the p-reductions of the f_i. Denote $d_+ = \sum_i d_i$. There is a deterministic algorithm to calculate the zeta function for X_p for all $p < N$ in*

$$\tilde{O}\left(2^{8n^2+17n+m+1} n^{4n+6} (d_+ + 2)^{4n^2+7n} N \log \|f\|\right) \text{ bit operations.}$$

Proof. This proceeds in much the same way as in Corollaries 11 and 12. The only extra item to keep track of is that there are at most N zeta functions produced[21], so the above must be repeated an additional factor of at most N times. We note that any prime being worked with is less than N, and $a = 1$, reducing Equation (1.8) to

$$\tilde{O}\left(N 2^{3m} \left(4d_+ + 5\right)^{2n} \log N\right) \text{ bit operations.}$$

This is dominated by the cost of producing the zeta functions. □

[21] This is a coarse bound. We'll do better later.

Chapter 2

Some Vital Combinatorial Identities

> "It is by will alone I set my mind in motion. It is by the juice of sapho that thoughts acquire speed, the lips acquire stains, the stains become a warning. It is by will alone I set my mind in motion."
>
> *David Lynch*, Dune (screenplay)

The basis of many of these results is a combinatorial identity that is independent of any sort of algebraic structure. This identity applies to any situation where a function maps between finite sets, and relates the size of the value set to a scaled sum of the sizes of various (set-wise) fiber product of the spaces involved.

Recall, the fiber product $X \times_Y X$ is defined as "the" set making the diagram in figure 2.1 commute (where "the" is used here because the construction is universal, that is if there were any other such sets and projection maps, the sets would be isomorphic in the category of sets), and the resulting diagram would commute.

For our purpose, we can generally just think of what it means for this diagram to commute in the category of sets, where we have

$$X \times_Y X = \{(x_1, x_2) \in X \times X : f(x_1) = f(x_2)\}.$$

$$\begin{array}{ccc} X \times_Y X & \xrightarrow{\pi_2} & X \\ {\scriptstyle \pi_1}\downarrow & & \downarrow{\scriptstyle f} \\ X & \xrightarrow[f]{} & Y \end{array}$$

Figure 2.1: Fiber Product (Universal Property not Depicted)

We similarly notate the k-iterated fiber product as

$$X^{\times_Y k} = \underbrace{X \times_Y \cdots \times_Y X}_{k \text{ terms}} = \left\{(x_1, \cdots, x_k) \in X^k : f(x_1) = \cdots = f(x_k)\right\}.$$

2.1 A Relationship Between the Fiber Product and the Cardinality of the Value Set

We count the number of points in these k-iterated fiber products (for any positive integer, k) up to some bound, namely the maximal number of points present in any fiber above any point in the image under the map f. More accessibly, the bound is a positive integer d so that $|f^{-1}(y)| \leq d$ for all $y \in Y$.

Theorem 14. *If X and Y are finite sets, and $f : X \to Y$ is a map such that any given fiber has at most d elements, then the cardinality of the image set of f is*

$$|V_f| = \sum_{i=1}^{d} (-1)^{i-1} N_i \sigma_i \left(1, \frac{1}{2}, \cdots, \frac{1}{d}\right), \tag{2.1}$$

where $N_k = \left|X^{\times_Y k}\right|$ and σ_i denotes the ith elementary symmetric polynomial on d elements.

Proof. Beginning in a similar way as Uchiyama [49] and Birch / Swinnerton-Dyer [4], we examine a family of subsets of V_f, namely

$$V_{f,i} = \left\{y \in V_f : \left|f^{-1}(y)\right| = i\right\}, \quad 1 \leq i \leq d.$$

Each element $\beta \in V_f$ must have at least one pre-image (as if β had no points in its pre-image, it would not be in the image!) and can have at most d points in its pre-image, so

$$V_f = \coprod_{1 \leq i \leq d} V_{f,i}$$

(where $\coprod$ denotes the disjoint union).

Continuing as in both Uchiyama [49] and Das [16], denote the cardinality of each of these sets as $m_i = |V_{f,i}|$. Any element in the image must have between 1 and d pre-images; we count elements in the image, grouped by the number of elements in the pre-image, yielding the equation

$$m_1 + \cdots + m_d = |V_f| \,. \tag{2.2}$$

Now, let

$$\tilde{N}_k = X^{\times_Y k}.$$

We are generally going to be more interested in the number of elements in such sets; we have already denoted this as $N_k = \left|\tilde{N}_k\right|$. We'll categorize the points in $\tilde{N}_k$ by their (shared) image.

We continue by counting the number of ways of forming each N_k in terms of the various m_i's. In particular, as each value in the image must be in exactly one of the $V_{f,i}$ sets, if $(x_1, \cdots, x_k) \in N_k$, then all of the x_i's in this k-tuple are pre-images of a value in the same $V_{f,i}$.

To illustrate the counting argument, we start with counting N_1: if $(\alpha_1) \in \tilde{N}_1$ with $f(\alpha_1) = \beta$, then β is in exactly one $V_{f,i}$ (as these sets partition V_f). There are m_1 distinct images in $V_{f,1}$, each of which has a distinct pre-image, so there are m_1 choices for (α_1) such that $\beta \in V_{f,1}$. If instead $\beta \in V_{f,2}$, then β could be one of m_2 distinct images, each of which has exactly 2 distinct pre-images, so there would be $2m_2$ choices for such an (α_1). Similarly, if $\beta \in V_{f,\ell}$, then there are m_ℓ distinct images, each of which have exactly ℓ distinct pre-images, so there would be exactly ℓm_ℓ choices for (α_1). There can be no overlap between each of these cases (as the $V_{f,i}$ partition V_f), so we can then sum and find $N_1 = m_1 + 2m_2 + \cdots + dm_d$.

For N_k, if $(\alpha_1, \cdots, \alpha_k) \in \tilde{N}_k$ with $f(\alpha_1) = \beta$ and $\beta \in V_{f,\ell}$, then there are m_ℓ distinct images, each of which have exactly ℓ distinct pre-images, so there would be exactly ℓm_ℓ choices for α_1, and ℓ choices for each of

$\alpha_2, \cdots, \alpha_k$, yielding a total of $\ell^k m_\ell$ choices for $(\alpha_1, \cdots, \alpha_k)$. Thus we see that in general

$$N_k = m_1 + 2^k m_2 + \cdots + d^k m_d. \tag{2.3}$$

Now, let us introduce a new variable, say $\xi = -|V_f|$. We can then rewrite (2.2) to be $m_1 + \cdots + m_d + \xi = 0$, and (2.3) to $m_1 + 2^k m_2 + \cdots + d^k m_d + 0\xi = N_k$ with $1 \le k \le d$; this system of equations yields

$$\begin{pmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & 2 & \cdots & d & 0 \\ 1 & 2^2 & \cdots & d^2 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 2^d & \cdots & d^d & 0 \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ \vdots \\ \xi \end{pmatrix} = \begin{pmatrix} 0 \\ N_1 \\ N_2 \\ \vdots \\ N_d \end{pmatrix}. \tag{2.4}$$

We then solve for ξ using Cramer's rule.

For Cramer's rule, we need two different determinants. First, we need the determinant of the $(d+1) \times (d+1)$ square matrix above, which we'll call A

Lemma 15.

$$\begin{aligned} \det A &= \det \begin{pmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & 2 & \cdots & d & 0 \\ 1 & 2^2 & \cdots & d^2 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 2^d & \cdots & d^d & 0 \end{pmatrix} \\ &= (-1)^d d!(d-1)!(d-2)! \cdots 2!1! \end{aligned}$$

Proof. For the determinant of A, we can expand along the last column and

then factor out the common terms from each column:

$$\begin{aligned}
\det A &= \det \begin{pmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & 2 & \cdots & d & 0 \\ 1 & 2^2 & \cdots & d^2 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 2^d & \cdots & d^d & 0 \end{pmatrix} \\
&= (-1)^{d+2} \det \begin{pmatrix} 1 & 2 & \cdots & d \\ 1 & 2^2 & \cdots & d^2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 2^d & \cdots & d^d \end{pmatrix} \\
&= (-1)^d d! \det \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & d \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 2^{d-1} & \cdots & d^{d-1} \end{pmatrix}.
\end{aligned}$$

This sub-matrix is (the transpose of) a Vandermonde matrix, so the determinant of the original matrix is:

$$\begin{aligned}
\det A &= (-1)^d d! \prod_{1 \le i < j \le d} (j - i) \\
&= (-1)^d d!(d-1)!(d-2)! \cdots 2!1!
\end{aligned}$$

□

We'll need the determinant of a new matrix for Cramer's rule. This new matrix, B, will be based on A, but with the last column replaced by the column vector on the right hand side of Equation (2.4). Calculating this determinant will require a modest effort.

Lemma 16.

$$\det B = \det \begin{pmatrix} 1 & 1 & \cdots & 1 & 0 \\ 1 & 2 & \cdots & d & N_1 \\ 1 & 2^2 & \cdots & d^2 & N_2 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 1 & 2^d & \cdots & d^d & N_d \end{pmatrix}$$
$$= (d-1)!(d-2)! \cdots 2!1! \sum_{i=1}^{d} (-1)^{d+i} N_i \sigma_{d-i}(1, 2, \cdots, d)$$

Proof. For the determinant of B, we have a somewhat similar looking determinant; again expanding along the last column:

$$\det B = \sum_{i=1}^{d} (-1)^{d+i} N_i M_{i+1,d+1},$$

where $M_{i+1,d+1}$ is the corresponding minor of B.

Each of these $M_{i+1,d+1}$ are "simple alternants".[40, Chapter XI][1, Chapter VI] We generalize B by replacing the elements $(1, \cdots, d)$ with a corresponding unique variable $(X_1, \cdots, X_d)$, producing $\hat{B}$, whose determinant can then be calculated in terms of the new minors of $\hat{M}_{i+1,d+1}$. We then write the (non-eliminated) powers as $\alpha = (\alpha_1, \cdots, \alpha_d)$, allowing us to denote:

$$\begin{aligned} a_\alpha &= \hat{M}_{i+1,d+1} \\ &= \det \left(X_n^{\alpha_m}\right)_{m,n=1}^{d} \text{ where } \alpha_m = \begin{cases} m-1 & 1 \le m < i+1 \\ m & i+1 \le m \le d \end{cases} \end{aligned}$$

We define $\delta = (0, 1, \cdots, d-1)$, then

$$a_\alpha = a_{\lambda+\delta} = \det \left(X_n^{\lambda_m + (m-1)}\right)_{n,m=1}^{d}$$

which forces $\lambda = (\underbrace{0, \cdots, 0}_{i \text{ terms}}, \underbrace{1, \cdots, 1}_{d-i \text{ terms}})$.[22]

[22] Stanley provides a wonderful introduction to these topics.[47, p. 335]

It is evident that if $X_m = X_n$ for any $m < n$ then a_α is 0. This implies that $(X_n - X_m)$ divides a_α for all $1 \leq m < n \leq d$, thus a_α is divisible by a_δ (the Vandermonde determinant). The quotient $a_{\lambda+\delta}/a_\delta$ (called a "bialternant") is the historical definition of the Schur polynomial of shape λ:

$$s_\lambda\left(X_1, \cdots, X_d\right) = \frac{a_{\lambda+\delta}\left(X_1, \cdots, X_d\right)}{a_\delta\left(X_1, \cdots, X_d\right)}.$$

Comparing this to the more standard combinatorial definition of the Schur polynomials:

$$s_\lambda = \sum_\beta K_{\lambda\beta} x^\beta$$

where β runs over all weak compositions (*i.e.*, start with an integer partition of $\ell = \sum_m \lambda_m$ padded with 0s to bring the partition length to the same length as λ. The set of weak compositions are every possible ordering of every such partition). Here, $K_{\lambda\beta}$ is the Kostka number, the number of semi-standard Young tableaux (SSYT) of shape λ and type β.

In this case, the form of λ causes all such tableaux to be a single column of length $d - i$; a tableau forms a valid SSYT only if the integers that fill the tableau strictly increase down the column. Each weak composition, β, establishes values that must be used to fill the tableau; there must be β_m total m's present in the tableau. As we are required to *strictly* increase down the column, this tells us that $K_{\lambda\beta} = 0$ for any β that contains any values other than 0 and 1, and there is exactly one way to arrange these numbers into our tableau: in increasing order. Thus

$$K_{\lambda\beta} = \begin{cases} 0 & \beta_m > 1 \text{ for any } i \\ 1 & \text{otherwise} \end{cases}$$

which suggests that each term in the sum s_λ has exactly $d-i$ distinct terms, and includes all possible arrangements. For this λ, we see that:

$$\begin{aligned} s_\lambda &= \sum_\beta K_{\lambda\beta} X^\beta \\ &= \sum_{1 \leq j_1 < j_2 < \cdots < j_{d-i} \leq d} X_{j_1} X_{j_2} \cdots X_{j_{d-i}} \\ &= \sigma_{d-i}\left(X_1, \cdots, X_d\right). \end{aligned}$$

That is, for this type of λ, s_λ is just the $(d-i)$th elementary symmetric polynomial on d variables, and thus

$$a_{\lambda+\delta}\left(X_1,\cdots,X_d\right)=\sigma_{d-i}\left(X_1,\cdots,X_d\right)a_\delta\left(X_1,\cdots,X_d\right).$$

We thus have:

$$\begin{aligned}M_{i+1,d+1}&=\sigma_{d-i}\left(1,\cdots,d\right)a_\delta\left(1,\cdots,d\right)\\&=\sigma_{d-i}\left(1,\cdots,d\right)\prod_{1\leq m<n\leq d}(n-m)\\&=\sigma_{d-i}\left(1,\cdots,d\right)(d-1)!(d-2)!\cdots 2!1!\end{aligned}$$

Finally combining these results,

$$\det B=(d-1)!(d-2)!\cdots 2!1!\sum_{i=1}^{d}(-1)^{d+i}N_i\sigma_{d-i}\left(1,2,\cdots,d\right).$$

□

Combining our results and applying Cramer's rule:

$$\begin{aligned}\xi&=\frac{\det B}{\det A}\\&=\frac{\left((d-1)!(d-2)!\cdots 2!1!\right)\sum_{i=1}^{d}(-1)^{d+i}N_i\sigma_{d-i}\left(1,2,\cdots,d\right)}{(-1)^d d!(d-1)!(d-2)!\cdots 2!1!}\\&=\frac{1}{d!}\sum_{i=1}^{d}(-1)^iN_i\sigma_{d-i}\left(1,2,\cdots,d\right)\\&=\sum_{i=1}^{d}(-1)^iN_i\sigma_i\left(1,\frac{1}{2},\cdots,\frac{1}{d}\right).\end{aligned}$$

Consequently, we have the desired result. □

As a small example, examine the map f in Figure 2.2.

A (tight) bound on the cardinality of any fiber for this map is 3. The most straight forward way of calculating N_k here is via Equation (2.3). The relevant quantities necessary for the calculation of Equation (2.1) are in Table 2.1.

Putting these together, we see that

$$|V_f|=5\cdot\frac{11}{6}-13\cdot 1+35\cdot\frac{1}{6}=2,$$

which is true by inspection of Figure 2.2.

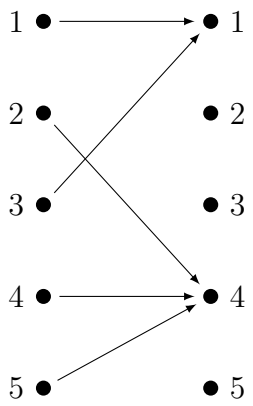

Figure 2.2: An Example Map, f

Table 2.1: Example Value Set Cardinality Calculation

j	m_j	N_j	$\sigma_j\left(1, \frac{1}{2}, \frac{1}{3}\right)$
1	0	5	11/6
2	1	13	1
3	1	35	1/6

2.2 A Note on the Calculation of Symmetric Polynomials

In the above sections, we see that we need $\sigma_i(X_1, \cdots, X_n)$, in particular for the case where $0 \leq i \leq n$, where $X_i = 1/i$. The definition of the elementary symmetric polynomials is

$$\prod_{i=1}^{n} (T + X_i) = e_0(X_1, \cdots, X_n)T^n + \cdots + e_n(X_1, \cdots, X_n). \tag{2.5}$$

Enumerating all the choices is trivial, but there are clearly 2^n choices, so if we proceeded naïvely, we expect any such approach to have computational complexity $\Omega(2^n)$ bit operations, which may impact our total complexity.

Lemma 17. *The values $\sigma_i(1, 1/2, \cdots, 1/d)$ for $0 \leq i \leq d$ can be computed in $\tilde{O}(d^5)$ bit operations*

Proof. We proceed by using Newton's identities,[38]

$$e_k(X_1, \cdots, X_n) = \frac{1}{k} \sum_{i=1}^{k} (-1)^{i-1} e_{k-i}(X_1, \cdots, X_n) p_i(X_1, \cdots, X_n), \quad (2.6)$$

where p_k is the kth power sum, that is

$$p_k(X_1, \cdots, X_n) = \sum_{j=1}^{n} X_j^k.$$

In order to calculate $\sigma_i(1, \cdots, 1/d)$ in this way, we need to first calculate the $p_i(1, \cdots, 1/d)$ (that is, the generalized harmonic number of order d of k). Note that if we denote $p_k(1, \cdots, 1/d)$ as p_k, and $t_i = d!/i$ then

$$\begin{aligned} p_k &= \frac{1}{1^k} + \frac{1}{2^k} + \cdots + \frac{1}{d^k} \\ &= \frac{(d!)^k + (d!)^k/2^k + \cdots + (d!)^k/d^k}{(d!)^k} \\ &= \frac{t_1^k + t_2^k + \cdots + t_d^k}{t_1^k}. \end{aligned}$$

As it happens, we need t_i^1 to t_i^d, so square-and-multiply exponentiation techniques aren't helpful. For the following, denote the number of bit operations required to multiply two n bit integers as $\mathrm{M}(n)$.

Lemma 18. *(Borwein) k integers, each of which can be stored in ℓ-bits, can be multiplied together in time complexity $O(\log k\, \mathrm{M}(k\ell))$ bit operations.*

Proof. This is the result of the standard recursive "split the input into two subproblems then combine" method.[6, Proposition 1] □

We assume that we operate using one of the many "fast" integer multiplication schemes, so $\mathrm{M}(n) = \tilde{O}(n)$ bit operations.

We first calculate $t_1 = d!$ using Lemma 18 in $\tilde{O}(d)$ bit operations, and the resulting value is stored in $O(d \log d)$ bits[23]. For each later i, we then divide by i, which again takes $\tilde{O}(d)$ bit operations, so the total time required

[23] Borwein's factorization-based method would do even better, but has the same soft-oh time complexity, so we don't bother.[6, Proposition 2]

to calculate all the necessary t_i values is $\tilde{O}(d^2)$ bit operations, and the size of t_i is $O(d \log d)$ bits.

Exponentiation occurs in the naïve way (as we need all the intermediate results), so for fixed i, we require d multiplications of integers no larger than $O(d^2 \log d)$ bits, so for fixed i this takes $\tilde{O}(d^3)$ bit operations. There are d choices for i, so we have a rough bound of $\tilde{O}(d^4)$ bit operations to calculate all of the $(t_i^k)_{i,j=1\ldots d}$. Each resulting p_k is a rational number which has both numerator and denominator of bit length $O(dk \log d)$ bits. Putting all d of the needed power sums into lowest terms then can occur in time $\tilde{O}(d^5)$ bit operations.

Now note that if we specialize Equation (2.5) to the case we are looking at and denote $e_k = e_k(1, \cdots, 1/d)$, then we see that

$$\prod_{i=1}^{n} \left(T + \frac{1}{i} \right) = e_0 T^n + e_1 T^{n-1} + \cdots + e_n.$$

Examining e_k, we find that a common denominator for the sum making up e_k is $d!$, and the numerator of each term is a product of $d-k$ values, the largest of which (associated with the choice of $1/1, 1/2, \cdots, 1/k$) is $d!/k!$. A bound for the numerator is thus

$$\binom{d}{k} \frac{d!}{k!} = \binom{d}{k}^2 \cdot (d-k)!.$$

We thus find that a bound for the bit length of the numerator is

$$\begin{aligned} \log \left(\binom{d}{k}^2 \cdot (d-k)! \right) &= O\left(k \log d + (d-k) \log(d-k) \right) \\ &= O\left((k+d) \log d \right) \\ &= O\left(d \log d \right) \text{ bits.} \end{aligned}$$

The denominator is similarly of length $\log(d!) = O(d \log d)$ bits.

Calculating e_k using Newton's identity then occurs in time complexity $O\left(k \, \mathrm{M}\left(dk \log d\right)\right)$ bit operations, which in soft-oh (with fast multiplication) is $\tilde{O}\left(dk^2\right)$ bit operations. This must be done d times, and $k \leq d$, so the entire sum for all $k \leq d$ is thus performed in time complexity $\tilde{O}(d^4)$ bit operations. Reducing them to lowest terms takes time complexity $\tilde{O}(d^5)$ bit operations, so the time complexity for calculating the full set of elementary symmetric polynomials is $\tilde{O}(d^5)$ bit operations. □

2.3 The Fiber Product and Fiber Signature

There's nothing about Cramer's rule that applies only to solving for $|V_f|$; we can just as reasonably use the same system to solve for any particular m_j in a very similar fashion.

These m_j provide a fairly complete view of how the map works with respect to its fibers. If (as above) m_j denotes the number of points in the image of a function f that have fibers of cardinality exactly j, and d denotes the the maximal fiber cardinality, then we'll refer to $M = (m_1, \cdots, m_d)$ as the *fiber signature of* f.

With these m_j, one can of course calculate $|V_f|$, but this is only a small portion of the combinatorial structure revealed. Note that for any particular $|V_f|$, any ordering of an integer partition of $|V_f|$ into d or fewer parts gives a possible fiber signature (and all of these are associated with functions!)

Given the above development, it is clear that one example structure revealed is the size of the k-iterated fiber product.

To calculate each m_j requires much the same process as in calculating $|V_f|$ above.

Theorem 19. *If X and Y are finite sets, and $f : X \to Y$ is a map such that any given fiber has at most d elements, then for any positive integer $j \leq d$, the number of points in the co-domain whose fiber has exactly j elements is*

$$m_j = \binom{d}{j} \frac{1}{j} \sum_{i=1}^{d} (-1)^{i+j} N_i \sigma_{i-1} \left(1, \frac{1}{2}, \cdots, \frac{1}{j-1}, \frac{1}{j+1}, \cdots, \frac{1}{d}\right),$$

where $N_k = \left|X^{\times_Y k}\right|$ and σ_i denotes the ith elementary symmetric polynomial on $d-1$ elements.

Proof. We start with Equation (2.4) and then apply Cramer's Rule.

Lemma 20. *If we let*

$$B_j = \det \begin{pmatrix} 1^0 & 2^0 & \cdots & (j-1)^0 & 0 & (j+1)^0 & \cdots & d^0 & 1 \\ 1^1 & 2^1 & \cdots & (j-1)^1 & N_1 & (j+1)^1 & \cdots & d^1 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 1^d & 2^d & \cdots & (j-1)^d & N_d & (j+1)^d & \cdots & d^d & 0 \end{pmatrix}$$

then $\det B_j$ *is of the form*

$$(d-1)!\cdots 1!\binom{d}{j}\sum_{i=1}^{d}(-1)^{i+j+d}N_i\sigma_{d-i}\left(1,\cdots,j-1,j+1,\cdots,d\right).$$

Proof. We start by expanding the determinant along the $(d+1)$th column, arriving at the moderately nicer

$$\begin{aligned}
&\det B_j\\
&=(-1)^{d+2}\det\begin{pmatrix}1^1 & \cdots & (j-1)^1 & N_1 & (j+1)^1 & \cdots & d^1\\ \vdots & & \vdots & \vdots & \vdots & & \vdots\\ 1^d & \cdots & (j-1)^d & N_d & (j+1)^d & \cdots & d^d\end{pmatrix}\\
&=(-1)^d\frac{d!}{j}\det\begin{pmatrix}1^0 & \cdots & N_1 & \cdots & d^0\\ \vdots & & \vdots & & \vdots\\ 1^{d-1} & \cdots & N_d & \cdots & d^{d-1}\end{pmatrix}.
\end{aligned}$$

Expanding this new matrix along the jth column results in minors of the form

$$C_{i,j}=\begin{pmatrix}1^0 & 2^0 & \cdots & (j-1)^0 & (j+1)^0 & \cdots & d^0\\ \vdots & \vdots & & \vdots & \vdots & & \vdots\\ 1^{i-2} & 2^{i-2} & \cdots & (j-1)^{i-2} & (j+1)^{i-2} & \cdots & d^{i-2}\\ 1^i & 2^i & \cdots & (j-1)^i & (j+1)^i & \cdots & d^i\\ \vdots & \vdots & & \vdots & \vdots & & \vdots\\ 1^{d-1} & 2^{d-1} & \cdots & (j-1)^{d-1} & (j+1)^{d-1} & \cdots & d^{d-1}\end{pmatrix}$$

whence we find $\det B_j$ is

$$\det B_j=(-1)^d\frac{d!}{j}\sum_{i=1}^{d}(-1)^{i+j}N_i\det C_{i,j}.$$

It simplifies our notation if we take $\Gamma=(1,\cdots,j-1,j+1,\cdots,d)$. As before, we have a bialternant of a very similar form. Here we have $\delta=(0,1,\cdots,d-2)$ and $\lambda=(\underbrace{0,\cdots,0}_{i-1\text{ terms}},\underbrace{1,\cdots,1}_{d-i\text{ terms}})$, yielding

$$\det C_{i,j}=\sigma_{d-i}\left(\Gamma\right)a_\delta\left(\Gamma\right).$$

This is slightly more complex, as the Vandermonde determinant is no longer a simple product of factorials. Using the notation of Lemma 16, we see that in particular,

$$\begin{aligned} a_\delta(\Gamma) &= \prod_{\substack{1\le u<v\le d \\ u,v\neq j}} (v-u) \\ &= \frac{(d-1)!}{(d-j)}\frac{(d-2)!}{(d-j-1)}\cdots\frac{j!}{1}\frac{(j-1)!}{(j-1)!}(j-2)!\cdots 2!1! \\ &= \frac{(d-1)!\cdots 1!}{(d-j)!(j-1)!}. \end{aligned}$$

Putting this together, we get the desired result. □

Thus, we can solve for $m_j = \frac{\det B_j}{\det A}$ (where $\det A$ was calculated in Lemma 15), and get:

$$m_j = \binom{d}{j}\frac{1}{d!}\sum_{i=1}^{d}(-1)^{i+j}N_i\sigma_{d-i}(\Gamma).$$

Distributing in the $\frac{1}{d!}$ term into the symmetric polynomial, we get products of i terms, each of the form $\frac{1}{k}$ $(1 \le k \le d)$, each with a $\frac{1}{j}$ term. Removing this common term, we are left with the desired result. □

We can then use this to calculate the complete fiber signature, which (at least in the category of sets) provides a large amount of information about a map between finite sets.

Proposition 21. *If X and Y are finite sets, and $f : X \to Y$ is a map such that any given fiber has at most d elements, and if N_1 to N_d are provided as input (where $N_k = \left|X^{\times_Y k}\right|$), then the necessary elementary symmetric polynomials can be computed in complexity*

$$\tilde{O}\left(d^6\right) \text{ bit operations,}$$

and the fiber signature of f can be computed in an additional

$$\tilde{O}\left(d^3 \log N_1\right) \text{ bit operations.}$$

Proof. In Lemma 17, we found that we could calculate all $\sigma_i(1, \cdots, 1/d)$ for all $0 \le i \le d$ in $\tilde{O}(d^5)$ bit operations; note that this is also a computational

upper bound for the case where we exclude one of the values and instead examine the elementary symmetric polynomials on $d-1$ variables (as is the case in Theorem 19). We can simply replace the excluded value with 0, and apply the same argument. We can thus calculate all the elementary symmetric polynomials required to calculate $(m_1, \cdots, m_d)$ in computational complexity $\tilde{O}(d^6)$ bit operations. As before, a bound on the length of the numerator and denominator of these rational numbers is $O(d \log d)$ bits.

Calculating all the necessary binomial coefficients can be done directly (using the falling factorial representation of the binomial coefficient) using Lemma 18 in computational complexity $\tilde{O}(d)$ bit operations; the resulting values are of length $O(d \log d)$ bits.

We are provided N_1 to N_d by hypothesis. Note that N_1 is the number of points in the full space (here called X), and a trivial bound for the size of later spaces is N_1^k, so each of these have a length bound of $O(d \log N_1)$ bits.

Each of the d multiplications (for fixed selection of j) occurs in time complexity $\tilde{O}(d \log N_1)$ bit operations, and there are d choices for j, so calculating all the series has time complexity $\tilde{O}(d^3 \log N_1)$ bit operations.

The entire operation thus has time complexity $\tilde{O}(d^6 + d^3 \log N_1)$ bit operations. □

Returning to our example map from Figure 2.2, we again tabulate the necessary information into Table 2.2.

Table 2.2: Example Value Set Cardinality Calculation

j	N_j	$\sigma_{j-1}\left(\frac{1}{2}, \frac{1}{3}\right)$	$\sigma_{j-1}\left(1, \frac{1}{3}\right)$	$\sigma_{j-1}\left(1, \frac{1}{2}\right)$
1	5	1	1	1
2	13	5/6	4/3	3/2
3	35	1/6	1/3	1/2

We can then find that

$$m_1 = \binom{3}{1} \cdot \frac{1}{1} \cdot \left(5 \cdot 1 - 13 \cdot \frac{5}{6} + 35 \cdot \frac{1}{6}\right) = 0$$
$$m_2 = \binom{3}{2} \cdot \frac{1}{2} \cdot \left(-5 \cdot 1 + 13 \cdot \frac{4}{3} - 35 \cdot \frac{1}{3}\right) = 1$$
$$m_3 = \binom{3}{3} \cdot \frac{1}{3} \cdot \left(5 \cdot 1 - 13 \cdot \frac{3}{2} + 35 \cdot \frac{1}{2}\right) = 1.$$

Which is the same fiber signature as seen in Table 2.1.

Chapter 3

Morphisms Between Affine Varieties over a Finite Field

> "Going beyond this point may result in death and/or loss of skiing privileges."
>
> Snow park boundary sign at Sierra Summit

With this underlying combinatorial and point counting framework in place, we can now proceed to describe algorithms for finding the value set cardinality and fiber signature of certain types of algebraic maps.

3.1 Notation

All of the results in this section use the following conventions and notation (or some specialization of it).

Let p be a prime, and a be a positive integer, with $q = p^a$. Let X and Y be algebraic varieties defined over $\mathbb{F}_q$.

More precisely, Let X be an affine variety over $\bar{\mathbb{F}}_q$ defined by the vanishing set of (a non-negative integer) ℓ polynomials in affine r-space[24]

$$\alpha_1(x_1, \cdots, x_r) = \cdots = \alpha_\ell(x_1, \cdots, x_r) = 0,$$

where each $\alpha_i \in \mathbb{F}_q[x_1, \cdots, x_r]$.

[24] As each of these polynomials provide constraints on the points in the variety, we operate under the convention that if $\ell = 0$, then $X = \mathbb{A}^r_{\mathbb{F}_q}$, and similarly for the variety Y.

Similarly, let Y be an affine variety over $\bar{\mathbb{F}}_q$ defined by the vanishing set of (a non-negative integer) m polynomials in affine s-space

$$\beta_1(y_1, \cdots, y_s) = \cdots = \beta_m(y_1, \cdots, y_s) = 0.$$

Denote the $\mathbb{F}_{q^w}$-rational points on X as $X(\mathbb{F}_{q^w})$, further denote $x = (x_1, \cdots, x_r)$, and the analogous notions for y.

Let f be a morphism from X to Y which is an s-tuple of polynomials $f(x) = (f_1(x), \cdots, f_s(x))$, where each $f_i \in \mathbb{F}_q[x_1, \cdots, x_r]$.

For notational convenience, denote

$$d_i = \begin{cases} \deg \alpha_i & i \leq \ell \\ \deg f_{i-\ell} & \ell < i \leq \ell + s \\ d_{i \pmod{\ell+s}+1} & \text{otherwise} \end{cases},$$

and denote the restriction $f|_{X(\mathbb{F}_{q^k})}$ as $f|_{q^k}$, which is evidently a function $f|_{q^k} : X(\mathbb{F}_{q^k}) \to Y(\mathbb{F}_{q^k})$.

We are interested in counting the value set of this morphism over $\mathbb{F}_q$, that is, we ask the question "if we view the domain of this map as the $\mathbb{F}_q$-rational points on X, what is the number of points in this map's value set?"

We'll start abstractly, and just assert that for some reason we may be able to bound the size of a fiber in some meaningful way, and see where that takes us.[25]

In order to motivate the (notationally unsightly) situation we find ourselves in, let's first examine an important restriction of this setting, namely the standard one-variable value set counting problem.

3.2 The One Dimensional Case

We examine the case where we are counting the value set of a one-variable polynomial over $\mathbb{F}_q$, which is the setting where this problem has been most

[25] In fact, we don't need the number of points in the fiber to be bounded, just the number of $\mathbb{F}_{q^r}$-rational points in the fiber above any $\mathbb{F}_{q^r}$-rational point, for suitable choice of r. This trivially *always* occurs (so these algorithms always work) but if the bound is too high, these algorithms perform worse than the naïve approach to counting the value set.

widely studied. We start by showing two naïve approaches to calculating this result, and then demonstrate the approach that we'll generalize to more general affine varieties. In the above notation, this is the situation where $\ell = m = 0$, and $r = s = 1$.

3.2.1 Naïve Algorithms

There are several naïve methods of calculating $|V_f|$. Perhaps the most obvious method is to evaluate the polynomial at each point in $\mathbb{F}_q$ and count how many unique images result. This approach uses q evaluations, each of which can be evaluated using the Horner scheme [31] in $2d - 1$ field multiplications, each in time complexity $O(a^{1+\lg 3} \lg^2 p)$ bit operations (here lg is the logarithm base 2), and d field additions, each in $O(a \lg p)$ bit operations.[26] The final counting can occur in time complexity $O(q)$ bit operations, which is negligible in comparison to the other operations.

Thus, our first naïve algorithm has time complexity $O(qda^{\lg 3-1} \lg^2 q)$ bit operations, or in "Soft-Oh" notation, $\tilde{O}(qd)$ bit operations. This algorithm is thus not polynomial in the input length, which in the dense polynomial model is assumed to be length $O(d \lg q)$ bits.

One can also approach this problem by operating on points in the co-domain. One has $f(x) = a$ for some $x \in \mathbb{F}_q$ if and only if $f_a(X) = f(X) - a$ has at least one linear factor. We can test for such factors by examining $\deg \gcd(f_a, X^q - X)$. This is computationally expensive for large q, so we instead examine $\deg \gcd(f_a, X^q - X \pmod{f_a})$, which is of the same degree.[27]

Multiplication of polynomials of degree no greater than d can occur in $O(\mathrm{M}(d))$ field operations, where $\mathrm{M}(d) = d \log d \log \log d$. Modular reduction then requires $O(\lg q \, \mathrm{M}(d))$ field operations, and the GCD calculation requires $O(\log d \, \mathrm{M}(d))$ field operations. Repeating this process at most q times identifies the entire image set, requiring $O(q \lg q \, \mathrm{M}(d))$ field multiplications. Combining, we get a computational complexity of

$$O(p^a da^{\lg 3-1} \lg^3 q \log d \log \log d) \text{ bit operations,}$$

[26] Estimates of bit operations for arithmetic operations in $\mathbb{F}_q$ assume an iterated extension approach.[3, p. 348]

[27] This is also the first step of Rabin's irreducibility test.[43]

or in "Soft-Oh" notation, $\tilde{O}(qd)$ bit operations.[28]

3.2.2 Value Set Cardinality via Point Counting

Theorem 14 gives us a way to express $|V_f|$ in terms of the number of points on a family of spaces on $\mathbb{F}_q^k$. If we had a way of getting N_k for $1 \leq k \leq d$, then we could calculate $|V_f|$.

We provide an independent proof of this corollary here for the purpose of exposition, but we will see that this corollary is a direct consequence of Theorem 23 in the next section.

We now proceed by counting points in the described spaces.

Corollary 22. *Let a be a positive integer, p be a prime, $q = p^a$, and $f(x) \in \mathbb{F}_q[x]$ be a polynomial with positive degree d. There is a deterministic algorithm that calculates the cardinality of the value set, $|V_f|$ in $\mathbb{F}_q$, and more generally the fiber signature of f, with computational complexity*

$$\tilde{O}\left(2^{6d-1}\lambda^{4d+3}d^{8d+1}a^2p^{1/2}\right) \text{ bit operation,}$$

where $\lambda = \max(a, \lceil (d+1)/2 \rceil)$.

Proof. In order to apply Theorem 14, we need to count the number of $\mathbb{F}_q$-rational points on $N_k = \left|\tilde{N}_k\right|$ with

$$\begin{aligned}\tilde{N}_k &= \left\{(x_1, \cdots, x_k) \in \mathbb{F}_q^k : f(x_1) = \cdots = f(x_k)\right\} \\ &= \left\{ (x_1, \cdots, x_k) \in \mathbb{F}_q^k \;\middle|\; \begin{array}{ll} f(x_1) - f(x_2) & = 0 \\ f(x_1) - f(x_3) & = 0 \\ & \vdots \\ f(x_1) - f(x_k) & = 0 \end{array} \right\}\end{aligned}$$

The above shows that $\tilde{N}_k$ is the simultaneous zero set for the $(k-1)$ polynomials $g_i(x_1, \cdots, x_k) = f(x_1) - f(x_{i+1})$, each of which is of degree d.

Each value of this one-variable polynomial can have at most d pre-images (as otherwise the polynomial translated by this image would have more than d roots!), so we can use Theorem 14 to calculate $|V_f|$ using Equation (2.1).

[28] If one was interested in estimating $|V_f|$, one could turn this algorithm into a probabilistic algorithm.[37]

We can calculate all the needed N_k values by applying Corollary 3, 4, 5, or 6 a total of $d-1$ times ($N_1 = q$, and need not be calculated).

These values are then scaled by evaluated elementary symmetric polynomials. All of the necessary elementary symmetric polynomials can be evaluated using Newton's identity in less than $\tilde{O}(d^5)$ bit operations (as shown in Lemma 17), which is dominated by the point counting operation. The bit lengths of the resulting denominators are bounded by $O(d \log d)$ bits. Each $N_k \leq q^k$, so the length of N_k is $O(ka \log p)$ bits, so the d multiplications and additions required to combine everything are dominated by the cost of the point counting calculation.

To summarize, $(d-1)$ invocations of any of the point counting algorithms for affine varieties would work in this case (the bound stated is associated with Corollary 6), by using the parameters in Table 3.1.

Table 3.1: Point Counting Parameters for Corollary 22

Parameter	Value
n	d
m	$d-1$
$\bar{d}$	$d+1$
d_+	$d(d-1)$

Note that the cost of calculating N_1 to N_d also dominates the cost to calculate the fiber signature as described in Proposition 21. □

As noted, the initial approach for Theorem 14 included an approach similar to Birch and Swinnerton-Dyer.[4] The difference is that they required that $x_i \neq x_j$ for $i \neq j$. The standard approach to representing such inequalities is the "Rabinowitsch trick". Using this trick, we introduce an additional variable, say y, and the additional equation

$$y \prod_{i<j} (x_j - x_i) = 1.$$

This is a degree $\binom{k}{2} + 1$ polynomial, which would increase the work factor of the algorithm dramatically.

3.3 The General Setting

Fundamentally, we will do the same proof as in Corollary 22 , but *more*.

Theorem 23. *Using the notation and conventions from Section 3.1, if there is a positive integer $\mathcal{D}$ so that $\left|(f|_q)^{-1}(y)\right| \leq \mathcal{D}$ for all $y \in Y(\mathbb{F}_q)$, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of f, with computational complexity*

$$\tilde{O}\left(2^{\mathcal{D}(\ell+s+r)-s}\mathcal{D}(\mathcal{D}r+2d_+\lambda+2\lambda)^{4\mathcal{D}r}\lambda^3 a^2 p^{1/2}\right) \text{ bit operations,}$$

where $\lambda = \max\left(a, \lceil(\mathcal{D}r+1)/2\rceil\right)$ and $d_+ = \sum_{i=1}^{\mathcal{D}\ell+(\mathcal{D}-1)s} d_i$.

Proof. We have (by hypothesis) bounded the size of any fiber, so we can directly apply Theorem 14 in this setting (just as in the case of a single polynomial.)

In particular, in this setting we have

$$\tilde{N}_k = X^{\times_Y k}.$$

This observation, along with Theorem 2.1, gives us the ability to connect the number of points in a twisted "diagonal" of the variety with the number of points in its value set.

Recall that $N_k = \left|\tilde{N}_k(\mathbb{F}_q)\right|$, and denote $\alpha(x) = (\alpha_1(x), \cdots, \alpha_\ell(x))$. We then construct the space (lightly abusing notation by requiring that "0" denote whatever sized zero vector is required to be sensible):

$$\begin{aligned}\tilde{N}_k(\mathbb{F}_q) &= \left\{\left(x^{(1)}, \cdots, x^{(k)}\right) \in X(\mathbb{F}_q)^k : f\left(x^{(1)}\right) = \cdots = f\left(x^{(k)}\right)\right\} \\ &= \left\{ \left(x^{(1)}, \cdots, x^{(k)}\right) \in \left(\mathbb{F}_q^r\right)^k \;\middle|\; \begin{array}{rl} \alpha(x^{(1)}) & = 0 \\ & \vdots \\ \alpha(x^{(k)}) & = 0 \\ f\left(x^{(1)}\right) - f\left(x^{(2)}\right) & = 0 \\ & \vdots \\ f\left(x^{(1)}\right) - f\left(x^{(k)}\right) & = 0 \end{array} \right\}\end{aligned}$$

This polynomial system is evidently in kr variables. Each α term represents ℓ distinct polynomials. Each f term represents s distinct polynomials. There are thus a total of $k\ell+(k-1)s$ total polynomials, each in kr variables.

By Theorem 14, we can calculate $|V_f|$ using Equation (2.1). We can calculate all the needed N_k values by applying Corollary 3, 4, 5, or 6 a total of $\mathcal{D}$ times in order to calculate N_k for $1 \leq k \leq d$. These values are then scaled by evaluated elementary symmetric polynomials. All of the necessary elementary symmetric polynomials can be evaluated using Newton's identity in less than $\tilde{O}(d^5)$ bit operations (as shown in Section 2.2), which is dominated by the point counting operation. The bit lengths of the resulting denominators are bounded by $O(d \log d)$ bits. Each $N_k \leq q^k$, so the bit length of N_k is $O(ka \log p)$ bits, so the d multiplications and additions required to combine everything are dominated by the cost of the point counting calculation.

To summarize, $\mathcal{D}$ invocations of any of the point counting algorithms for affine varieties would work in this case (the bound stated is associated with Corollary 6), by using the parameters in Table 3.2.

Table 3.2: Point Counting Parameters for Theorem 23

Parameter	Value
n	$\mathcal{D}r$
m	$\mathcal{D}(\ell + s) - s$
$\bar{d}$	$\max_{1 \leq i \leq s+\ell} d_i$
d_+	$\sum_{i=1}^{\mathcal{D}\ell+(\mathcal{D}-1)s} d_i$

For the calculation of the fiber signature, note that in this case $N_1 \leq q^r$, so the cost of calculating the fiber signature of f given by Proposition 21 is bounded by $\tilde{O}(d^6 ra \log p)$ bit operations, which is dominated by the cost of calculating $N_1, \cdots, N_d$.

□

It is instructive to note that if we examine the polynomial case (that is, fix $\ell = m = 0$, $\mathcal{D} = d$, $r = s = 1$), we get Corollary 22, so Corollary 22 is actually a corollary of Theorem 23.

3.3.1 Applications

We will principally be interested in the case where certain algebraic structure is maintained by our morphism, namely we need f to be a dominant *finite* morphism. Recall that a dominant morphism $f : X \to Y$, where X

and Y are affine varieties, is finite if and only if $\mathcal{O}(X)$ is a finitely generated $\mathcal{O}(Y)$-module via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism

$$f^\star : \mathcal{O}(Y) \to \mathcal{O}(X).$$

We extract a bound from a standard result from algebraic geometry.[29]

Lemma 24. *If $f : X \to Y$ is a finite dominant morphism and $\mathcal{O}(X)$ is generated by t elements or fewer as an $\mathcal{O}(Y)$-module (via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism $f^\star$), then $|f^{-1}(y)| \leq t$ for all $y \in Y$. If X is irreducible, then the fibers of f have cardinality at most the degree of f.*

This gives us the bounds necessary to establish the following two corollaries:

Corollary 25. *Using the notation and conventions from Section 3.1, if X is irreducible and f is a finite dominant morphism from X to Y of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity described in Theorem 23, with $\mathcal{D} = d$.*

If instead, we start with affine varieties that are not irreducible we may still be able to say something so long as $\mathcal{O}(X)$ are finitely generated over $\mathcal{O}(Y)$ (via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism $f^\star$). Let t be a bound on the number of elements in such a generating set. We then have that $|f^{-1}(y)| \leq t$, so we can apply the same proof as in Corollary 25 (but using the bound t instead of d), which leads to the following result.

Corollary 26. *Using the notation and conventions from Section 3.1, if f is a finite dominant morphism, and $\mathcal{O}(X)$ is generated by a set of t elements from $\mathcal{O}(Y)$ (via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism $f^\star$), then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity described in Theorem 23, with $\mathcal{D} = t$.*

One important special case of the above is

[29] This lemma is included for reference. Reasonable proof of this lemma is available in most variety-oriented algebraic geometry texts, *e.g.*, the course notes by S. Paul Smith.[46, §1.10]

Corollary 27. *Using the notation and conventions from Section 3.1, if f is a finite dominant morphism from $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ to $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_q$, and more generally the fiber signature of $f|_q$, with computational complexity*

$$\tilde{O}\left(2^{2dr-r} d(dr + 2d_+\lambda + 2\lambda)^{4dr} \lambda^3 a^2 p^{1/2}\right) \text{ bit operations,}$$

where $\lambda = \max\left(a, \lceil (dr+1)/2 \rceil\right)$ and $d_+ = \sum_{i=1}^{(d-1)r} d_i$.

Proof. The ring of regular functions for $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$, $\bar{\mathbb{F}}_q[x_1, \cdots, x_r]$, is an integral domain, so $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ is irreducible. Lemma 24 and Theorem 23 (with $l = m = 0$, $s = r$, and $\mathcal{D} = d$) then give us the desired result. □

Chapter 4

Amortized Cost of Counting the Value Set

> "PhD thesis protip: your committee will only read the first eight and last three pages. Just fill the middle part with Duran Duran lyrics."
>
> *Professor Matthew D. Green*, Johns Hopkins University

> "Searching for the undeniable truth that a man is just a fool."
>
> *Duran Duran*, New Religion

Exploring these questions on a singleton basis isn't the only way to proceed, of course. By analogy, the question of counting $\mathbb{F}_q$-rational points on a variety has been expanded to examining the behavior of this count as we vary the characteristic of the field or the degree of the extension.[44]

In this way, we may be interested in the behavior of the cardinality of the value set of a particular polynomial as we vary the degree of the extension of the field, or as we change the characteristic of the field.

In this section, we'll denote the value set of the $f|_{q^r}$ as $V_f(\mathbb{F}_{q^r})$.

The insight that enables this approach is that the polynomials that define the spaces $\tilde{N}_k$ (when viewed correctly) don't change as we vary these parameters. As such, we can start in just the same way as in Section 3.2.2.

4.1 Amortized Cost in Fixed Characteristic

Theorem 28. *Using the notation and conventions from Section 3.1, and additionally letting R be a positive integer, if there is a positive integer $\mathcal{D}$ so that $\left|(f|_{q^R})^{-1}(y)\right| \leq \mathcal{D}$ for all $y \in Y(\mathbb{F}_{q^R})$, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity*

$$\tilde{O}\left(2^{\mathcal{D}(8\mathcal{D}r^2+17r+\ell+s)-s}\mathcal{D}^{4\mathcal{D}r+5}r^{4\mathcal{D}r+4}\left(d_+ + 2\right)^{\mathcal{D}r(4\mathcal{D}r+7)}a^{4\mathcal{D}r+4}p^{1/2} + R^2a\mathcal{D}^2r2^{\mathcal{D}\ell+(\mathcal{D}-1)s}\left(4d_+ + 5\right)^{\mathcal{D}r}\log p\right) \text{ bit operations,}$$

where $d_+ = \sum_{i=1}^{\mathcal{D}\ell+(\mathcal{D}-1)s} d_i$.

Proof. The first part of this proof proceeds exactly as in Theorem 28.

We have (by hypothesis) bounded the size of any fiber, so we can again apply Theorem 14 and calculate $|V_f|$ using Equation (2.1).

$$\begin{aligned}\tilde{N}_k &= X^{\times_Y k} \\ &= \left\{(x^{(1)}, \cdots, x^{(k)}) \in X^k : f\left(x^{(1)}\right) = \cdots = f\left(x^{(k)}\right)\right\}.\end{aligned}$$

Denote $N_{k,w} = \left|\tilde{N}_k(\mathbb{F}_{q^w})\right|$, $\alpha(x) = (\alpha_1(x), \cdots, \alpha_\ell(x))$ and the space

$$\begin{aligned}\tilde{N}_k(\mathbb{F}_{q^w}) &= \left\{\left(x^{(1)}, \cdots, x^{(k)}\right) \in X(\mathbb{F}_{q^w})^k : f\left(x^{(1)}\right) = \cdots = f\left(x^{(k)}\right)\right\} \\ &= \left\{ \left(x^{(1)}, \cdots, x^{(k)}\right) \in \left(\mathbb{F}_{q^w}^r\right)^k \;\middle|\; \begin{array}{rl} \alpha(x^{(1)}) & = 0 \\ & \vdots \\ \alpha(x^{(k)}) & = 0 \\ f\left(x^{(1)}\right) - f\left(x^{(2)}\right) & = 0 \\ & \vdots \\ f\left(x^{(1)}\right) - f\left(x^{(k)}\right) & = 0 \end{array} \right\}.\end{aligned}$$

This polynomial system is evidently in kr variables. Each α term represents ℓ distinct polynomials. Each f term represents s distinct polynomials. There are thus a total of $k\ell+(k-1)s$ total polynomials, each in kr variables.

We now specialize to dealing with the zeta functions, rather than a distinct point counting algorithm.

Note that $N_{k,w}$ changes as we vary w, but the underlying polynomials that define the space do not, thus the variety $\tilde{N}_k$ does not change. As such, for fixed k, the values $N_{k,w}$ can be extracted from the logarithmic derivative of the zeta function. We calculate one zeta function per $\tilde{N}_k$ using corollaries 11 or 12, and then calculate all the needed $N_{k,w}$ values by applying Lemma 7.

The computational complexity of this algorithm is presented using Corollary 12, but if Corollary 11 were used instead, each of the $\mathcal{D}$ zeta function calculations would occur using the parameters in Table 4.1.

Table 4.1: Zeta Function Calculation Parameters for Theorem 28

Parameter	Value
n	$\mathcal{D}r$
m	$\mathcal{D}(\ell+s)-s$
d_+	$\sum_{i=1}^{\mathcal{D}\ell+(\mathcal{D}-1)s} d_i$

All of the resulting zeta functions are rational. Denote the zeta function associated with $\tilde{N}_k$ as $g(T)/h(T)$. Again using the degree bound that we found in the proof of Corollaries 11 and 12, for an affine space in n variables and defined by m polynomials, the degree of both the numerator and denominator of the zeta function is bounded by

$$D \leq 2^m \left(4d_+ + 5\right)^n .$$

Thus, in our case, we have a degree bound of

$$D \leq 2^{k\ell+(k-1)s} \left(4d_+ + 5\right)^{kr} .$$

Again using our coefficient bound, for the n-variable case, we find that the coefficients of both g and h are bounded by q^{Dn}, so

$$\begin{aligned} B &= O\left(Dkra \log p\right) \\ &= O\left(akr2^{k\ell+(k-1)s} \left(4d_+ + 5\right)^{kr} \log p\right) \text{ bits.} \end{aligned}$$

In order to extract the $N_{k,w}$, we now apply Lemma 7, and find that we can recover the first R values for $N_{k,w}$ in

$$\tilde{O}\left(R^2 akr2^{k\ell+(k-1)s} \left(4d_+ + 5\right)^{kr} \log p\right) \text{ bit operations.}$$

By hypothesis, f can have at most $\mathcal{D}$ pre-images, so by Theorem 14, we can calculate $|V_f|$ using Equation (2.1), which requires $\mathcal{D}$ iterations of the above, the most costly of which is $k = \mathcal{D}$. As before, calculation of the elementary symmetric polynomial is not an impediment (in this case particularly so, as the value of the symmetric polynomials need only be calculated once!)

For ease of book keeping, we can arrange the resulting values into a matrix operation:

$$\begin{pmatrix} N_{1,1} & N_{2,1} & \cdots & N_{\mathcal{D},1} \\ N_{1,2} & N_{2,2} & \cdots & N_{\mathcal{D},2} \\ \vdots & \vdots & \ddots & \vdots \\ N_{1,R} & N_{2,R} & \cdots & N_{\mathcal{D},R} \end{pmatrix}_{R\times\mathcal{D}} \begin{pmatrix} \sigma_1\left(1, \frac{1}{2}, \cdots, \frac{1}{\mathcal{D}}\right) \\ -\sigma_2\left(1, \frac{1}{2}, \cdots, \frac{1}{\mathcal{D}}\right) \\ \vdots \\ (-1)^{\mathcal{D}-1}\sigma_{\mathcal{D}}\left(1, \frac{1}{2}, \cdots, \frac{1}{\mathcal{D}}\right) \end{pmatrix} = \begin{pmatrix} |V_f(\mathbb{F}_{q^1})| \\ |V_f(\mathbb{F}_{q^2})| \\ \vdots \\ \left|V_f(\mathbb{F}_{q^R})\right| \end{pmatrix}.$$

As seen in Lemma 17, the cost to calculate the elementary symmetric polynomials is dominated by the rest of the calculation, and the matrix operation occurs in $\tilde{O}(\mathcal{D}Rra\log p)$ bit operations, also dominated by the cost of the rest of the operation.

For calculation of the fiber signatures, referring to Proposition 21, we need only compute the elementary symmetric polynomials once for the entire computation, with complexity $\tilde{O}(d^6)$ bit computations.

Using the trivial bound

$$\log N_{1,w} \leq Rra\log p,$$

we see that the cost to calculate the R distinct fiber signatures is bounded by $\tilde{O}(\mathcal{D}^3R^2ra\log p)$ bit operations, which is dominated by the cost of extracting the $N_{k,w}$ values from the zeta function. □

Specializing to the case of one variable polynomials, we can let $s = r = 1$, $\ell = 0$, $\mathcal{D} = d$, which leaves us with the following corollary.

Corollary 29. *Let a and R be positive integers, p be a prime, $q = p^a$, and f be a polynomial $f(x) \in \mathbb{F}_q[x]$, of positive degree d. There is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity*

$$\tilde{O}\left(2^{8d^2+18d-1}d^{8d^2+18d+5}a^{4d+4}p^{1/2} + R^2 2^{3d-1}d^{2d+2}a\log p\right) \text{ bit operations.}$$

It is useful to think about this after fixing the polynomial that is being evaluated, and then looking at the cost of calculating the cardinality of a single value set (amortized across the total number of value sets counted), at which point the computational complexity is

$$\tilde{O}\left(a^{4d+4}p^{1/2} + R^2 a \log p\right) \text{ bit operations.}$$

Note that if R grows suitably quickly as compared to q, then this gives an algorithm whose amortized cost (per value set cardinality calculated) is (soft-)polynomial with respect to input, and (soft-)linear with respect to output.

We can also apply Theorem 28 in the case of morphisms with the right structure, as we have seen in Corollaries 25 and 26.

Corollary 30. *Using the notation introduced in Theorem 28, if X is irreducible and f is a finite dominant morphism from X to Y of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity described in Theorem 28, with $\mathcal{D} = d$.*

Corollary 31. *Using the notation introduced in Theorem 28, if f is a finite dominant morphism, and $\mathcal{O}(X)$ is generated by a set of t elements from $\mathcal{O}(Y)$ (via the induced $\bar{\mathbb{F}}_q$-algebra homomorphism $f^\star$), then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity described in Theorem 28, with $\mathcal{D} = t$.*

One important special case of the above (as in Corollary 27) can be found by letting $l = m = 0$, $s = r$, and $\mathcal{D} = d$, whence we arrive at

Corollary 32. *Using the notation introduced in Theorem 28, if f is a finite dominant morphism from $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ to $\mathbb{A}^r_{\bar{\mathbb{F}}_q}$ of fixed degree d, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{q^w}$, and more generally the fiber signature of $f|_{q^w}$, for all $w \leq R$ with computational complexity*

$$\tilde{O}\Big(2^{d(8dr^2+18r)-r} d^{4dr+5} r^{4dr+4} \left(d_+ + 2\right)^{dr(4dr+7)} a^{4dr+4} p^{1/2} + R^2 a d^2 r 2^{(d-1)r} \left(4d_+ + 5\right)^{dr} \log p\Big) \text{ bit operations,}$$

where $d_+ = \sum_{i=1}^{(d-1)r} d_i$.

4.2 Amortized Cost Across Many Characteristics

In much the same way as the above, we can examine $|V_f(\mathbb{F}_{p^a})|$ as we vary both p and a. One important difference here (that will somewhat simplify the statement of our results) is that in the affine case we can't expect any particular polynomial (or s-tuple of polynomials) to remain a morphism as we vary p if we restrict the regular functions on the space. As such, we abandon some of the generality above, and concentrate on the instance where we are dealing with a morphism from $\mathbb{A}^r$ to $\mathbb{A}^s$.

We again revisit the notation that we are using.

Let f be an s-tuple of polynomials over the integers, $f(x) = (f_1(x), \cdots, f_s(x))$, where $f_i \in \mathbb{Z}[x_1, \cdots, x_r]$ is of total degree d_i. Denote the maximal coefficient (in absolute value) of f_i as $\|f_i\|$.

For each prime p we can consider the p-reduction of f, denoted f_p, by reducing the coefficients of the polynomials modulo p and considering the resulting map as a morphism $f_p : \mathbb{A}^r_{\mathbb{F}_p} \to \mathbb{A}^s_{\mathbb{F}_p}$. We are interested in characterizing the cardinality of the value set of such f_p once we restrict the domain to some finite field of characteristic p.

We could use Theorem 28 to accomplish this task for fixed p, and calculate the cardinality of the value set for all finite extensions of $\mathbb{F}_p$ less than or equal to degree R. The following results instead allow us to simultaneously perform this calculation for all primes p less than some bound N and for all field extensions above $\mathbb{F}_q$ of degree less than or equal to R.

For notational convenience, denote the restriction $f_p|_{X(\mathbb{F}_{p^w})}$ as $f|_{p^w}$, which is evidently a function $f|_{p^w} : \mathbb{F}^r_{p^w} \to \mathbb{F}^s_{p^w}$.

Theorem 33. *Let r, s, N and R be positive integers. Let f be an s-tuple of polynomials $f(x) = (f_1(x), \cdots, f_s(x))$, where $f_i(x) = \mathbb{Z}[x_1, \cdots, x_r]$, where the total degree of f_i is d_i.*

If there is a positive integer $\mathcal{D}$ so that $\left| \left(f|_{p^R} \right)^{-1} (y) \right| \leq \mathcal{D}$ for all $y \in \mathbb{F}^s_{p^R}$ and for all primes $p < N$, then there is a deterministic algorithm to calculate the cardinality of the value set of $f|_{p^w}$, and more generally the fiber signature

of $f|_{p^w}$*, for all* $w \leq R$ *and all primes* $p < N$*, with computational complexity*

$$\tilde{O}\left(2^{\mathcal{D}(8\mathcal{D}r^2+17r+s)-s+1}\mathcal{D}^{4\mathcal{D}r+8}r^{4\mathcal{D}r+6}((\mathcal{D}-1)d_+ + 2)^{\mathcal{D}r(4\mathcal{D}r+7)}N\log\|f\| + N\mathcal{D}^2R^2r2^{(\mathcal{D}-1)s}\left(4(\mathcal{D}-1)d_+ + 5\right)^{\mathcal{D}r}\right) \text{ bit operations,}$$

where $d_+ = \sum_{i=1}^{s} d_i$ *and* $\|f\| = \prod_{j=1}^{s}\|f_j\|$.

Proof. We have (by hypothesis) bounded the size of any fiber, so we can again apply Theorem 14 and can calculate $|V_f|$ using Equation (2.1).

For each choice of prime p, use X_p to denote affine r-space over $\bar{\mathbb{F}}_p$ and Y_p to denote affine s-space over $\bar{\mathbb{F}}_p$. We again then have

$$\tilde{N}_{k,p} = X_p^{\times_{Y_p} k}.$$

Denote $N_{k,p,w} = \left|\tilde{N}_{k,p}(\mathbb{F}_{p^w})\right|$. We then use Corollary 13 to calculate the zeta functions for $\tilde{N}_{k,p}$ for all primes $p \leq N$ for each value of k from 1 to $\mathcal{D}$. We can then extract all the $N_{k,p,w}$ values for all $w \leq R$ by invoking Lemma 7 on the zeta functions for $Z_{\tilde{N}_{k,p}}$.

The variety $\tilde{N}_{k,p}$ is described by the $k-1$ total s-tuples of polynomials, of the form

$$\begin{cases} f\left(x^{(1)}\right) - f\left(x^{(2)}\right) &= 0 \\ \vdots & \\ f\left(x^{(1)}\right) - f\left(x^{(k)}\right) &= 0 \end{cases}$$

As such, there are thus a total of $(k-1)s$ polynomials, and the system is in kr variables.

We can now apply Corollary 13 a total of $\mathcal{D}$ times (once for each $k = 1 \ldots \mathcal{D}$), which has time complexity bounded by

$$\tilde{O}\left(2^{\mathcal{D}(8\mathcal{D}r^2+17r+s)-s+1}\mathcal{D}^{4\mathcal{D}r+8}r^{4\mathcal{D}r+6}((\mathcal{D}-1)d_+ + 2)^{\mathcal{D}r(4\mathcal{D}r+7)}N\log\|f\|\right) \text{ bit operations.}$$

This results in $\mathcal{D}$ zeta functions for each of the $\pi(N)$ distinct primes less than or equal to N. The resulting zeta functions are rational, and as in Theorem 28, the degree of the numerator and denominator are bounded by

$$D \leq 2^{(\mathcal{D}-1)s}\left(4(\mathcal{D}-1)d_+ + 5\right)^{kr}.$$

We then apply our coefficient bound, and find that the maximal coefficient is less than N^{krD}, so our length bound is

$$B = kr2^{(\mathcal{D}-1)s}\left(4(\mathcal{D}-1)d_+ + 5\right)^{kr}\log N.$$

For fixed k and p, we can extract $N_{k,p,w}$ from the zeta functions (for all positive integers w less than or equal to R) by application of Lemma 7. We apply this lemma $\mathcal{D}\pi(N)$ times to extract all the necessary values of $N_{k,p,w}$, which requires

$$\tilde{O}\left(\pi(N)\mathcal{D}^2R^2r2^{(\mathcal{D}-1)s}\left(4(\mathcal{D}-1)d_+ + 5\right)^{\mathcal{D}r}\log N\right) \text{ bit operations.}$$

Indexing the primes less than N as $p_1, \cdots, p_{\pi(N)}$ and letting $\Gamma = (1, 1/2, \cdots, 1/\mathcal{D})$, we can form the final step of the calculation into a matrix operation:

$$\begin{pmatrix} N_{1,p_1,1} & \cdots & N_{\mathcal{D},p_1,1} \\ N_{1,p_1,2} & \cdots & N_{\mathcal{D},p_1,2} \\ \vdots & \ddots & \vdots \\ N_{1,p_1,R} & \cdots & N_{\mathcal{D},p_1,R} \\ N_{1,p_2,1} & \cdots & N_{\mathcal{D},p_2,1} \\ \vdots & \ddots & \vdots \\ N_{1,p_{\pi(N)},R} & \cdots & N_{\mathcal{D},p_{\pi(N)},R} \end{pmatrix} \begin{pmatrix} \sigma_1(\Gamma) \\ -\sigma_2(\Gamma) \\ \vdots \\ (-1)^{\mathcal{D}-1}\sigma_{\mathcal{D}}(\Gamma) \end{pmatrix} = \begin{pmatrix} |V_f(\mathbb{F}_{p_1})| \\ \left|V_f(\mathbb{F}_{p_1^2})\right| \\ \vdots \\ \left|V_f(\mathbb{F}_{p_1^R})\right| \\ |V_f(\mathbb{F}_{p_2})| \\ \vdots \\ \left|V_f(\mathbb{F}_{p_{\pi(N)}^R})\right| \end{pmatrix}.$$

As seen previously, the cost to calculate the elementary symmetric polynomials is dominated by the rest of the calculation, and the matrix operation occurs in $\tilde{O}(R^2\pi(N)\mathcal{D}\log N)$ bit operations, which is also dominated by the cost of the rest of the operation.

To calculate the fiber signatures, referring to Proposition 21, we need only compute the elementary symmetric polynomials once for the entire computation, with complexity $\tilde{O}(d^6)$ bit operations.

There are a total of $\pi(N)R$ sets of N values. Using the trivial bound

$$\log N_{1,p_{\pi(N)},R} \leq Rr\log N,$$

we see that the cost to calculate the $\pi(N)R$ distinct fiber signatures is bounded by $\tilde{O}(N\mathcal{D}^3R^2r)$ bit operations, which is dominated by the cost of extracting the $N_{k,p_i,w}$ values from the zeta function. □

Specializing to the case of one-variable polynomials, we can let $s = r = 1$, $\mathcal{D} = d$, which leaves us with the following corollary.

Corollary 34. *Let N and R be positive integers and f be a polynomial $f(x) \in \mathbb{Z}[x]$ of positive degree d. There is a deterministic algorithm to calculate the cardinality of the value set of $f|_{p^w}$, and more generally the fiber signature for $f|_{p^w}$, for all positive integers $w \leq R$ and for all primes $p \leq N$ with computational complexity*

$$\tilde{O}\left(2^{d(8d+18)} d^{8d^2+18d+8} N \log \|f\| + NR^2 2^{3d-1} d^{2d+2}\right) \text{ bit operations.}$$

From this, we get a total of $\pi(N)R$ value set (and fiber signature) results. For large N, we have

$$\pi(N) \sim \frac{N}{\log N}.$$

We again fix the polynomial, and examine the cost per value set counted (amortizing the cost over these $\pi(N)R$ values), resulting in an amortized computational complexity of

$$\tilde{O}(R \log N) \text{ bit operations.}$$

In this way, we have an algorithm that is (soft-)polynomial in the size of the underlying field. In particular, this (partially) resolves Wan's conjecture affirmatively: this algorithm counts the value set in (amortized) cost polynomial in $\log q$.

Compare this computational complexity with the computational complexity of Corollaries 22 or 29 (again assuming we fix the polynomial f), summarized in Table 4.2 (here letting $p = N$ and $a = R$). This shows that this "doubly amortized" approach is a marked improvement over either of the prior approaches.

Table 4.2: Comparison of Amortized Complexities (fixed polynomial)

Cor.	Complexity (bit operations)
22	$\tilde{O}\left(R^2 N^{1/2}\right)$
29	$\tilde{O}\left(R^{-1} N^{1/2} + R \log N\right)$
34	$\tilde{O}\left(R \log N\right)$

Chapter 5

Conclusion

> "Beware of the man who works hard to learn something, learns it, and finds himself no wiser than before. He is full of murderous resentment of people who are ignorant without having come by their ignorance the hard way."
>
> *Kurt Vonnegut*, Cat's Cradle

5.1 Findings, Redux

We adapted and analyzed existing zeta function calculation algorithms so that we could apply their results in our setting for both point counting and calculating zeta functions in affine varieties over finite fields. We also developed an algorithm that extracts the number of $\mathbb{F}_{q^k}$-rational points on a variety from that variety's zeta function, for all positive k less than some bound R.

We found a pair of combinatorial results that provide a link between the number of elements in the k-iterated fiber product via a map f and the value set of the map, or more generally its fiber signature. In instances where we can provide a suitably low bound and suitably efficient point counting algorithms exist, these links can be used to provide efficient algorithms for calculating the value set or the fiber signature.

In particular, we provide algorithms for calculating the value set of a morphism for varieties over a finite field in two cases where we can bound

the number of points in the fiber. This specializes to the case of the long-standing problem of calculating the cardinality of the value set (or more generally the fiber signature) of a polynomial over a finite field.

These findings also lead to two sets of related algorithms, where the cost of a much larger calculation can be amortized over the number of results calculated, with a better resulting cost per value set (or fiber signature) calculated. In both cases, the zeta functions for the k-iterated fiber products of the spaces were used to extract the number of $\mathbb{F}_{q^k}$-rational points for all finite extensions of $\mathbb{F}_q$ of degree up to some bound. This data was then combined to solve for the cardinality of the value set and fiber signature for these maps over many extensions of the base finite field. In the first case, this is done for a single characteristic. In the second case, an algorithm (due to Harvey) that computes zeta functions across many characteristics is used as the basis of a similar approach, where this same process is used across both many extensions and many characteristics.

This latter approach yields an amortized cost of counting the value set of a map that is polynomial in $\log q$, partially resolving a conjecture of Wan.

5.2 Future Work

There are several ways to extend these results. The underlying combinatorial relationship applies to any case where the size of the fiber of a map can be bounded; there are surely many different instances where this should be possible, particularly in the case where we only care about $\mathbb{F}_q$-rational points. As such, one approach to extending these results would be to locate additional settings where such a bound is possible.

It may be possible to apply Harvey's general methods to the case of function fields; one approach here would be to use Lauder's general approach with smooth, projective hypersurfaces[33], but replace Lauder's cohomological tools with Harvey's (p-adic and Witt vector based) tools.

It seems likely that the fiber signature can be used to directly calculate many types of important information regarding the map; many combinatorial results should be extractable from this fiber signature.

The existence of these algorithms (and their computational complexity) provides some information about some asymptotic results regarding the value set. In particular, it should be possible to gain further insight into the impact of the degree of the polynomial on the asymptotic error term.

Bibliography

[1] A.C. Aitken. *Determinants and Matrices*. Oilver and Boyd, ninth edition, 1959.

[2] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, New York, NY, USA, 1st edition, 2009.

[3] Eric Bach and Jeffrey Shallit. *Algorithmic Number Theory*, volume 1. The MIT Press, 1997.

[4] Bryan John Birch and Henry Peter Francis Swinnerton-Dyer. Note on a problem of Chowla. *Polska Akademia Nauk. Instytut Matematyczny. Acta Arithmetica*, 5:417–423, 1959.

[5] Enrico Bombieri. On exponential sums in finite fields, II. *Inventiones mathematicae*, 47(1):29–39, 1978.

[6] Peter B. Borwein. On the complexity of calculating factorials. *Journal of Algorithms*, 6(3):376–380, 1985. DOI: 10.1016/0196-6774(85)90006-9 . URL `http://dx.doi.org/10.1016/0196-6774(85)90006-9`.

[7] Peter J. Cameron. *Combinatorics: Topics, Techniques, Algorithms*. Cambridge University Press, 1994.

[8] Leonard Carlitz. On the number of distinct values of a polynomial with coefficients in a finite field. *Proceedings of the Japan Academy*, 31:119–120, 1955.

[9] Leonard Carlitz, Donald John Lewis, William H. Mills, and Ernst Gabor Straus. Polynomials over finite fields with minimal value sets. *Mathematika. A Journal of Pure and Applied Mathematics*, 8:121–130, 1961.

[10] Qi Cheng, Joshua E. Hill, and Daqing Wan. Counting value sets: algorithm and complexity. In Everett W. Howe and Kiran S. Kedlaya, editors, *ANTS X: Proceedings of the Tenth Algorithmic Number Theory Symposium*, pages 235–248. Mathematical Sciences Publishers, 2013. URL http://msp.org/obs/2013/1-1/obs-v1-n1-p12-s.pdf.

[11] Wun Seng Chou, Javier Gomez-Calderon, and Gary L. Mullen. Value sets of Dickson polynomials over finite fields. *Journal of Number Theory*, 30(3): 334–344, 1988. DOI: 10.1016/0022-314X(88)90006-6. URL http://dx.doi.org/10.1016/0022-314X(88)90006-6.

[12] Stephen D. Cohen. The distribution of polynomials over finite fields. *Polska Akademia Nauk. Instytut Matematyczny. Acta Arithmetica*, 17:255–271, 1970.

[13] Thomas W. Cusick. Value sets of some polynomials over finite fields GF(2^{2m}). *SIAM Journal on Computing*, 27(1):120–131 (electronic), 1998. DOI: 10.1137/S0097539794270352. URL http://dx.doi.org/10.1137/S0097539794270352.

[14] Thomas W. Cusick. Polynomials over base 2 finite fields with evenly distributed values. *Finite Fields and their Applications*, 11(2):278–291, 2005. DOI: 10.1016/j.ffa.2004.10.001. URL http://dx.doi.org/10.1016/j.ffa.2004.10.001.

[15] Thomas W. Cusick and Peter Müller. Wan's bound for value sets of polynomials. In *Finite fields and applications (Glasgow, 1995)*, volume 233 of *London Math. Soc. Lecture Note Ser.*, pages 69–72. Cambridge Univ. Press, Cambridge, 1996. DOI: 10.1017/CBO9780511525988.008. URL http://dx.doi.org/10.1017/CBO9780511525988.008.

[16] Pinaki Das. The number of polynomials of a given degree over a finite field with value sets of a given cardinality. *Finite Fields and their Applications*, 9 (2):168–174, 2003. DOI: 10.1016/S1071-5797(02)00020-5. URL http://dx.doi.org/10.1016/S1071-5797(02)00020-5.

[17] Leonard Eugene Dickson. The analytic representation of substitutions on a power of a prime number of letters with a discussion of the linear group. *Annals of Mathematics*, 11(1-6):65–120, 1896/97. DOI: 10.2307/1967217. URL http://dx.doi.org/10.2307/1967217.

[18] Bernard Dwork. On the rationality of the zeta function of an algebraic variety. *American Journal of Mathematics*, 82:631–648, 1960.

[19] Joachim von zur Gathen. Values of polynomials over finite fields. *Bulletin of the Australian Mathematical Society*, 43(1):141–146, 1991. DOI: 10.1017/S0004972700028860. URL http://dx.doi.org/10.1017/S0004972700028860.

[20] Joachim von zur Gathen. Tests for permutation polynomials. *SIAM Journal on Computing*, 20(3):591–602, 1991. DOI: 10.1137/0220037. URL http://dx.doi.org/10.1137/0220037.

[21] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, second edition, 2003.

[22] Javier Gomez-Calderon. *Polynomials with small value set over finite fields*. PhD thesis, The University of Arizona, 1986.

[23] Javier Gomez-Calderon. A note on polynomials with minimal value set over finite fields. *Mathematika. A Journal of Pure and Applied Mathematics*, 35(1):144–148, 1988. DOI: 10.1112/S0025579300006355. URL http://dx.doi.org/10.1112/S0025579300006355.

[24] Javier Gomez-Calderon and Daniel J. Madden. Polynomials with small value set over finite fields. *Journal of Number Theory*, 28(2):167–188, 1988. DOI: 10.1016/0022-314X(88)90064-9. URL http://dx.doi.org/10.1016/0022-314X(88)90064-9.

[25] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley, second edition, 1989.

[26] Alexander Grothendieck. Formule de Lefschetz et rationalité des fonctions L. In *Séminaire Bourbaki, Vol. 9*, pages Exp. No. 279, 41–55. Soc. Math. France, Paris, 1964.

[27] David Harvey. Computing zeta functions of arithmetic schemes. *ArXiv e-prints*, February 2014.

[28] David R. Hayes. A geometric approach to permutation polynomials over a finite field. *Duke Mathematical Journal*, 34:293–305, 1967.

[29] Richard Kantor. Über die Anzahl inkongruenter Werte ganzer, rationaler Funktionen. *Monatshefte für Mathematik und Physik*, 26(1):24–39, 1915. DOI: 10.1007/BF01999438. URL http://dx.doi.org/10.1007/BF01999438.

[30] Neeraj Kayal. Recognizing permutation functions in polynomial time. *Electronic Colloquium on Computational Complexity (ECCC)*, TR05(008), 2005. URL http://eccc.hpi-web.de/eccc-reports/2005/TR05-008/index.html.

[31] Donald Knuth. *The Art of Computer Programming*, volume 2. Addison-Wesley, third edition, 1998.

[32] Michiel Kosters. *Groups and fields in arithmetic*. PhD thesis, University of Leiden, 2014. URL http://pub.math.leidenuniv.nl/~kostersmf/PhD/Kosters_binnenwerk_3.pdf.

[33] Alan G. B. Lauder. Rigid cohomology and p-adic point counting. *Journal de Théorie des Nombres de Bordeaux*, 17(1):169–180, 2005.

[34] Alan G. B. Lauder and Daqing Wan. Counting points on varieties over finite fields of small characteristic. In J.P. Buhler and P. Stevenhagen, editors, *Algorithmic Number Theory*, pages 579 – 612. Cambridge University Press, 2008.

[35] Rudolf Lidl and Harald Niederreiter. *Finite Fields*. Cambridge University Press, 1997.

[36] Keju Ma and Joachim von zur Gathen. The computational complexity of recognizing permutation functions. *Computational Complexity*, 5(1):76–97, 1995. DOI: 10.1007/BF01277957. URL http://dx.doi.org/10.1007/BF01277957.

[37] Keju Ma and Joachim von zur Gathen. Tests for permutation functions. *Finite Fields and their Applications*, 1(1):31–56, 1995. DOI: 10.1006/ffta.1995.1003. URL http://dx.doi.org/10.1006/ffta.1995.1003.

[38] D. G. Mead. Newton's identities. *The American Mathematical Monthly*, 99 (8):pp. 749–751, 1992.

[39] William H. Mills. Polynomials with minimal value sets. *Pacific Journal of Mathematics*, 14:225–241, 1964.

[40] Thomas Muir and William H. Metzler. *A Treatise on the Theory of Determinants*. Dover Publications, Inc., 1960.

[41] Gary L. Mullen. Permutation polynomials over finite fields. In *Finite fields, coding theory, and advances in communications and computing (Las Vegas, NV, 1991)*, volume 141 of *Lecture Notes in Pure and Appl. Math.*, pages 131–151. Dekker, New York, 1993.

[42] Gary L. Mullen, Daqing Wan, and Qiang Wang. Value sets of polynomial maps over finite fields. *Quarterly Journal of Mathematics*, 64(4):1191 – 1196, 2013.

[43] Michael O. Rabin. Probabilistic algorithms in finite fields. *SIAM Journal on Computing*, 9(2):273–280, 1980. DOI: 10.1137/0209024. URL http://dx.doi.org/10.1137/0209024.

[44] Jean-Pierre Serre. *Lectures on N_X(p)*. Research Notes in Mathematics. CRC Press, Taylor & Francis Group, 2012.

[45] Igor E. Shparlinski. A deterministic test for permutation polynomials. *Computational Complexity*, 2(2):129–132, 1992. DOI: 10.1007/BF01202000. URL http://dx.doi.org/10.1007/BF01202000.

[46] S. Paul Smith. Affine algebraic geometry. 2006. URL https://www.math.washington.edu/~smith/Teaching/504/alggeom.pdf.

[47] Richard P. Stanley. *Enumerative Combinatorics*, volume 2. Cambridge University Press, 2001.

[48] Zhi-Wei Sun. On value sets of polynomials over a field. *Finite Fields and their Applications*, 14(2):470–481, 2008. DOI: 10.1016/j.ffa.2007.05.002. URL http://dx.doi.org/10.1016/j.ffa.2007.05.002.

[49] Saburô Uchiyama. Sur le nombre des valeurs distinctes d'un polynôme à coefficients dans un corps fini. *Proceedings of the Japan Academy*, 30:930–933, 1954. URL http://www.untruth.org/~josh/math/translations/uchiyama.html.

[50] Saburô Uchiyama. Note on the mean value of $V(f)$. *Proceedings of the Japan Academy*, 31:199–201, 1955.

[51] J. Felipe Voloch. On the number of values taken by a polynomial over a finite field. *Polska Akademia Nauk. Instytut Matematyczny. Acta Arithmetica*, 52 (2):197–201, 1989.

[52] Daqing Wan. A p-adic lifting lemma and its applications to permutation polynomials. In *Finite fields, coding theory, and advances in communications and computing (Las Vegas, NV, 1991)*, volume 141 of *Lecture Notes in Pure and Appl. Math.*, pages 209–216. Dekker, New York, 1993.

[53] Daqing Wan. Algorithmic theory of zeta functions over finite fields. In *Algorithmic number theory: lattices, number fields, curves and cryptography*, volume 44 of *Math. Sci. Res. Inst. Publ.*, pages 551–578. Cambridge Univ. Press, Cambridge, 2008.