Counting Value Sets

Algorithm and Complexity

Qi Cheng[1], **Joshua E. Hill**[2], Daqing Wan[2]

[1]School of Computer Science, The University of Oklahoma

[2]Department of Mathematics, University of California, Irvine

ANTS X

July 13, 2012

Talk Outline

Introduction Outline

1. Introduction and History
   - Asymptotic Results
   - Exact Results
   - Prior Complexity Results
2. Our Results
3. Conclusion

The Problem

- Let $f \in \mathbb{F}_{p^m}[x]$, of degree $d > 0$.
- Denote the value set $V_f = \{f(\alpha) \mid \alpha \in \mathbb{F}_{p^m}\}$.
- We are interested in the cardinality of V_f, which we denote $\#(V_f)$.

Cardinality of Image Sets

$$\left\lceil \frac{q}{d} \right\rceil \leq \#\left(V_f\right) \leq q$$

- These bounds are sharp!
- If $\#\left(V_f\right) = \left\lceil \frac{q}{d} \right\rceil$, then f is a polynomial with a minimal value set.
- If $\#\left(V_f\right) = q$, then f is a permutation polynomial.

Subsection 1

Asymptotic Results

The Shape of the Problem (Average Results)

A vital companion function:

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

- If $f^*(u,v)$ is absolutely irreducible $\#\left(V_f\right) > \frac{q}{2}$ for sufficiently large p [Uchiyama 1954]
- On average $\#\left(V_f\right) \sim \mu_d q + O_d(1)$ with μ_d is the series $1 - e^{-1}$ truncated at d terms. [Uchiyama 1955]

Asymptotic Results I

$$\#\left(V_f\right) = \mu q + O_d(\sqrt{q})$$

First asymptotic results [Birch and Swinnerton-Dyer, 1959]

- μ is dependent on some Galois groups induced by f. (more later)
- The case where f is a "general polynomial" then $\mu = \mu_d$.

Asymptotic Results II

Cohen gave a way to explicitly calculate μ [Cohen, 1970]

- Let K be the splitting field for $f(x) - t$ over $\mathbb{F}_q(t)$.
- Denote $k' = K \cap \bar{\mathbb{F}}_q$.
- $G^*(f) = \{\sigma \in G(f) \mid K_\sigma \cap k' = \mathbb{F}_q\}$.
- $G_1(f) = \{\sigma \in G(f) \mid \sigma \text{ fixes at least one point}\}$.
- $G_1^*(f) = G_1(f) \cap G^*(f)$.

$$\mu = \frac{\#\left(G_1^*\right)}{\#\left(G^*\right)}$$

- This provides a wonderful combinatorial explanation of μ_d (It is the proportion of non-derangements!)

Subsection 2

Exact Results

Exact Results

Exact values for $\#\left(V_f\right)$ are known for very few classes of polynomials:

1. Permutation polynomials (and exceptional polynomials)
2. Polynomials with a minimal (or very small) value set
3. Other

Permutation Polynomials

The class of polynomials where $\#\left(V_f\right) = q$

- These polynomials are uncommon (density $\sim e^{-q}$ for large q).
- Dickson found all of the permutation polynomials with $d \leq 6$. [Dickson 1896]

Exceptional Polynomials

Hayes harmonized these apparently disparate results by casting this into an Algo-Geometric setting [Hayes 1967]

Definition

$f(X) \in \mathbb{F}_q[X]$ is an exceptional polynomial if, when $f^*(X, Y)$ is factored into irreducibles over $\mathbb{F}_q[X, Y]$, all of these irreducible factors are not absolutely irreducible (that is, each irreducible factor cannot be irreducible over $\bar{\mathbb{F}}_q[X, Y]$.)

- All exceptional polynomials are permutation polynomials. [Cohen 1970], [Wan, 1993]
- If $d > 1$, $p \nmid d$ and $q > d^4$, then all permutation polynomials are exceptional polynomials (by the Lang-Weil Bound).

Small Image Set Polynomials

- All polynomials with minimal value sets with $d \leq \sqrt{q}$ were characterized in [Carlitz, Lewis, Mills, Straus 1961, and Mills 1964].
- All polynomials with $d^4 < q$ and $\#\left(V_f\right) < 2q/d$ were characterized in [Gomez-Calderon, 1986].
- Polynomials whose minimal value sets form subfields of $\mathbb{F}_q$ were characterized in [Borges-Conceição, 2012].

Other Cases

$\#\left(V_f\right)$ is known in a few other cases:

- The degree 0 and 1 cases are clear.
- The degree 2,3 cases are due to [Kantor 1915] and [Uchiyama 1955].
- For p-linear polynomials, $\#\left(V_f\right)$ is known due to linearity.
- Dickson Polynomials [Chou, Gomez-Calderon, Mullen 1988].
- $f(x) = x^k(1+x)^{2^m-1}$ in $\mathbb{F}_{2^m}$ (for $k = \pm 1, \pm 2, 4$) and $f(x) = (x+1)^d + x^d + 1$ for particular values of d [Cusick 2005].

An Important Note

- These results may seem to suggest that V_f can only be of certain forms. This is completely false.
- Lagrange interpolation can be used to build a polynomial with any desired image set.
- The restrictions discussed tell us that the choice of degree is not independent of the size of the image set.

Subsection 3

Prior Complexity Results

Partial Solutions

One can view the problem of finding $\#\left(V_f\right)$ as being a generalization of the problem of determining if a polynomial, f, is a permutation polynomial. There are a few algorithms for this:

- A deterministic permutation polynomials test was provided in [Shparlinski, 1992] which runs in $\tilde{O}((dq)^{6/7})$.
- The connection between exceptional and permutation polynomials was used in [Ma, von zur Gathen, 1995] to provide a zero-error probabilistic polynomial-time (ZPP) algorithm running in $\tilde{O}(d \log q)$.
- An approach relying on the classification of exceptional polynomials (which in turn relies on the classification of finite simple groups) is developed in [Kayal, 2005], which provides a deterministic-polynomial-time test running in $(d \log q)^{O(1)}$.

Naïve Algorithms

How to calculate $\#\left(V_f\right)$?

- Evaluate f at each point in $\mathbb{F}_q$. Cost: $\tilde{O}(qd)$ bit operations.
- For each $a \in \mathbb{F}_q$, $a \in V_f \Leftrightarrow \deg \gcd(f(x) - a, X^q - X) > 0$. Cost: $\tilde{O}(qd)$ bit operations.

Results Outline

Subsection 1

Complexity

When I Use a Word...

Are there polynomial-time algorithms for computing $\#\left(V_f\right)$? But *how* do we represent this polynomial?

A polynomial in:

- dense representation includes all coefficients up to the polynomial's degree (even those that are zero).
- sparse representation includes only the non-zero coefficients, along with the degree of the corresponding terms.
- straight-line program is defined recursively, as $x_1 = \alpha$, $x_2 = x$, and then $x_i = x_j \odot x_k$ where α is chosen so that $\mathbb{F}_q = \mathbb{F}_p[\alpha]$, $0 \leq j, k < i$, and $\odot$ is $+, -, \times$.

Complexity Classes

Decision Problem Complexity

- P is the complexity class of all decision problems that can be solved in polynomial time.
- NP is the complexity class of decision problem whose positive solutions can be verified in polynomial time.

Counting Problem Complexity

- #P (read: *"sharp-P"*) is the set of counting problems whose corresponding decision problem is in NP.
- #P-hard is the computational class of counting problems that all #P problems can be reduced to using a polynomial-time counting reduction.

Complexity Results I

Theorem

The problem of counting the value set of a sparse polynomial over a finite field of characteristic $p = 2$ is #P-hard.

We show this by providing a polynomial-time reduction of a 3SAT formula with n variables and m clauses to a sparse polynomial over $\mathbb{F}_{2^{n+m}}$; the cardinality of the value set of this polynomial reveals the number of satisfying assignments of the 3SAT formula.

Complexity Results II

Theorem

If the polynomial over $\mathbb{F}_p$ is given as a straight-line program, then the problem of counting the value set is #P-hard under RP-reduction.

We show this by providing a randomized-polynomial-time reduction of the counting subset sum problem (SSP) to counting the value set of a constructed polynomial as a straight-line program.

Subsection 2

Algorithm

$\#\left(V_f\right)$ and Point Counting

Another connection between $\#\left(V_f\right)$ and an algo-geometric structure:

Proposition

If $f \in \mathbb{F}_q[x]$ of positive degree d, then

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

where $N_k = \#\left(\left\{(x_1, \ldots, x_k) \in \mathbb{F}_q^k \mid f(x_1) = \cdots = f(x_k)\right\}\right)$ and σ_i is the ith elementary symmetric function on d elements.

Proof Outline I

- $V_{f,i} = \left\{x \in V_f \mid \#\left(f^{-1}(x)\right) = i\right\}$ with $1 \leq i \leq d$ forms a partition of V_f.
- Let $m_i = \#\left(V_{f,i}\right)$. Thus $m_1 + \cdots + m_d = \#\left(V_f\right)$. Introduce a new value $\xi = -\#\left(V_f\right)$. We then have:

$$m_1 + \cdots + m_d + \xi = 0$$

- Define the space $\tilde{N}_k = \left\{(x_1, \ldots, x_k) \in \mathbb{F}_q^k \mid f(x_1) = \cdots = f(x_k)\right\}$. Then $N_k = \#\left(\tilde{N}_k\right)$.
- By a counting argument,

$$m_1 + 2^k m_2 + \cdots + d^k m_d = N_k$$

Proof Outline II

Arrange this into a system of equations:

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

Solve for ξ using Cramer's rule. (Warning: some determinant magic)

Variations on a Theme of Matrices

You can just as reasonably solve for m_j through the same process:

Proposition

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

Counting Points...

- This equation is in terms of N_k, which we must calculate.
- $\tilde{N}_k$ isn't of any particularly desirable form: in particular, we can't assume that it is non-singular projective or an abelian variety (if it were, faster algorithms would apply!)
- We'll proceed through trickery.

Point Counting Algorithm

Some notation first. If $f \in \mathbb{F}_q[x_1, \ldots, x_n]$, let the variety X be the zeros of f over $\bar{\mathbb{F}}_q$. Denote $X(\mathbb{F}_{q^k}) = X \cap \mathbb{F}_{q^k}$.
We'll need the point counting algorithm of Lauder and Wan [Lauder-Wan 2008]:

Lemma

If f has total degree d in n variables and $p = O((d \log q)^C)$ for some constant C, then $\#\left(X(\mathbb{F}_{q^k})\right)$ can be calculated in polynomial time.

Algorithm for finding $\#\left(V_f\right)$

Theorem

There is an explicit polynomial R and a deterministic algorithm which, for any $f \in \mathbb{F}_q[x]$ (with $q = p^m$, p a prime, f degree d), calculates $\#\left(V_f\right)$. This algorithm requires a number of bit operations bounded by $R(m^d d^d p^d)$.

More explicit performance: $\tilde{O}\left(2^{8d+1} m^{6d+4} d^{12d-1} p^{4d+2}\right)$ bit operations. In particular, for fixed d and suitably small p (i.e., $p = O((d \log q)^C)$ for some constant C), this is polynomial time (for dense representation of the polynomial).

Proof Outline

Define:

$$F_k(\mathbf{x}, \mathbf{z}) = z_1 \left(f(x_1) - f(x_2) \right) + \cdots + z_{k-1} \left(f(x_1) - f(x_k) \right)$$

- If $\boldsymbol{\gamma} \in \tilde{N}_k$ then $F_k(\boldsymbol{\gamma}, \mathbf{z}) = 0$.
- If $\boldsymbol{\gamma} \in \mathbb{F}_q^k \setminus \tilde{N}_k$ then the solutions to $F_k(\boldsymbol{\gamma}, \mathbf{z})$ form a $(k-2)$-dimensional subspace of $\mathbb{F}_q^{k-1}$.
- If we denote the number of solutions to $F_k(\mathbf{x}, \mathbf{z})$ as $\#\left(F_k\right)$, then we have
$$\#\left(F_k\right) = q^{k-1} N_k + q^{k-2}(q^k - N_k)$$
- So, we can solve:
$$N_k = \frac{\#\left(F_k\right) - q^{2k-2}}{q^{k-2}(q-1)}$$
- And that's it!

Conclusion Outline

Summary

The problem of counting the value set of a polynomial:

- in sparse representation over $\mathbb{F}_{2^k}$ is in #P-hard.
- as a straight-line program over $\mathbb{F}_p$ is in #P-hard (under RP-reduction).
- in dense representation over $\mathbb{F}_q$ is in P for fixed d and sufficiently small p.

Conjecture

The problem of counting the value set of a polynomial in dense representation over $\mathbb{F}_q$ is in P for fixed d (for any p).

That's All Folks!

Thank You!