Counting Polynomials over finite fields and Analysis of Algorithms

Daniel Panario
University of Toronto

October 7, 1996

[summary by Mireille Régnier]

1. Motivation

In this talk, we comment on several problems in finite fields, and their relation with analytic combinatorics. Algebraic algorithms that deal with polynomials over finite fields can often be analyzed by counting polynomials with particular properties. We show that the most important characteristics of these algorithms can be treated systematically by a methodology based on generating functions and asymptotic analysis. We focus on three problems: polynomial factorization, irreducibility tests for polynomials, and discrete logarithm. For each problem, we present an efficient algorithm, we derive interesting counting expressions, and we mention known results.

2. Basic methodology

2.1. **Generating functions.** Let Φ be a class of monic polynomials, χ some integer-valued parameter on Φ. Let

$$\Phi(z,u) = \sum_{\omega \in \Phi} z^{|\omega|} u^{\chi(\omega)}.$$

The coefficient $[z^n u^k]\Phi(z,u)$ represents the number of polynomials of degree n with χ-parameter equal to k. Averages and standard deviations are obtained by taking successive derivatives of bivariate generating functions with respect to u, then setting $u = 1$. For instance, the mean is:

$$\frac{[z^n] \left.\frac{\partial P(z,u)}{\partial u}\right|_{u=1}}{[z^n]P(z,1)} = \frac{p_n'(1)}{p_n(1)}.$$

2.2. **Asymptotic analysis.** Generating functions encode exact informations on their coefficients. Their behavior near their dominant singularity is an important source of coefficient asymptotics.

A first method is known as singularity analysis due to Flajolet & Odlyzko. This requires analytic continuation (isolated singularity). However, there are some problems in which the generating functions present a natural boundary at $|z| = 1$ (each point at the unit circle is singular). Darboux's method could be used as an alternative in these cases. Finally, in some cases we use also a saddle point approximation.

2.3. **Permutation model.** The joint distribution of degrees in the prime decomposition of a random polynomial over $\mathbb{F}_q$ having degree n admits as a limit, when $q \to \infty$ (n staying fixed!), the joint distribution of cycle lengths in random permutations of size n. This gives rise to a useful heuristic: *probabilistic properties of polynomial factorization often have a shape resembling that of corresponding properties of the cycle decomposition of permutations to which they usually reduce as* $q \to \infty$.

3. Factoring polynomials over finite fields

The results in this part of the talk are from [1]. The Polynomial factorization algorithm proceeds in three steps:

ERF: *Elimination of repeated factors* replaces a polynomial by square-free ones that contain all the irreducible factors of the original polynomial with exponents reduced to 1.

DDF: *Distinct-degree factorization* splits a square-free polynomial into a product of polynomials whose irreducible factors all have the same degree.

EDF: *Equal-degree factorization* factors a polynomial whose irreducible factors have the same degree.

As our interest is in *dominant asymptotics*, we restrict our attention to the costs of products and gcd's that we assume to have constant costs τ_1 and τ_2 respectively.

3.1. **Elimination of repeated factors (ERF).** The first step in the factorization chain of a polynomial is the *elimination of repeated factors* (ERF). One proves that:

Theorem 1. *(i) A random polynomial of degree $n \geq 2$ in $\mathbb{F}_q[x]$ has a probability $1 - 1/q$ to be square-free.*

(ii) The degree of the non-square-free part of a random polynomial has expected value asymptotic to

$$C_q = \sum_{n \geq 1} \frac{nI_n}{q^{2n} - q^n},$$

where I_n is the number of irreducible polynomials of degree n, and a geometrically decaying probability tail. When $q \to \infty$, then $C_q \sim 1/q$.

Consequently, the overall cost of the recursive calls in the elimination of repeated factors remains $O(1)$ on average; alternative strategies giving the full square-free factorization will lead to asymptotically equivalent costs; the ERF phase has a cost dominated by that of its first gcd.

Theorem 2. *The expected cost of the ERF phase applied to a random polynomial of degree n satisfies*

$$\overline{ERF}_n \sim \tau_2 n^2.$$

3.2. **Distinct-degree factorization (DDF).** The second stage of our factorization algorithm requires finding the *distinct-degree factorization* (DDF) of the square-free polynomial a, i.e., splitting a under the form $b_1 \cdots b_n$ where b_k is the product of irreducible factors of a of degree k. The algorithm is $O(n^3)$. We provide a precise comparison of three strategies for the DDF phase: the basic rule, the "half-degree" rule and the "early abort" rule. The global saving of the early abort rule is of 36%, and the expected cost of $O(\log q \cdot n^3)$ for DDF clearly dominates the whole factorization chain.

3.3. **Equal-degree factorization (EDF).** DDF does not completely factor a polynomial that has different factors of same degree.

Theorem 3. *(i) The probability that DDF yields the complete factorization is asymptotic to*

$$c_q = \prod_{n \geq 1} \left(1 + \frac{I_n}{q^n - 1}\right) (1 - q^{-n})^{I_n},$$

$$c_2 \doteq 0.6656,\ c_{257} \doteq 0.5618,\ c_\infty = e^{-\gamma} \doteq 0.5614.$$

(ii) The degree of the part of the polynomial that remains to be factored by the EDF algorithm is asymptotic to $\log n$.

The factorization problem is reduced to factoring polynomials b_k that have all their irreducible factors of the same (known) degree k. Our reference is Cantor-Zassenhaus' probabilistic algorithm.

Each factor of b has probability $\alpha = (q-1)/(2q)$ to be a factor of d, and probability $\beta = (q+1)/(2q)$ to divide b/d. A random choice splits b in $\langle \ell, j-\ell \rangle$ factors with Bernoulli probability $\binom{j}{\ell}\alpha^\ell \beta^{j-\ell}$. The analysis combines a recursive partitioning problem akin to digital tries with estimates on the degree of irreducible factors of random polynomials.

Theorem 4. *The expected cost of the EDF phase satisfies*

$$\overline{EDF}_n \sim \frac{\tau_1}{\alpha\beta} \sum_{k=1}^{[n/2]} \mu_k, \qquad \mu_k = \left\lfloor \log_2(q^k-1)/2 \right\rfloor + \nu(q^k-1)/2 - 1.$$

In addition, this cost is $O(n^2)$, and for $-1/3 \le \xi_n \le 1/3$,

$$\overline{EDF}_n \sim \left(\frac{3}{4}\tau_1 \frac{q^2}{q^2-1} \log_2 q \cdot n^2 \right) (1 + \xi_n + o(1)).$$

4. Irreducibility tests for polynomials

A fundamental problem in finite fields is the construction of extension fields, that may be done by using an irreducible polynomial over the ground field with degree equal to the degree of the extension. Therefore, finding irreducible polynomials is a central problem in finite fields. A probabilistic algorithm is presented in [5]. The central idea is to take polynomials at random and test them for irreducibility. This suggests the study of the probability that a random polynomial of degree n contains no irreducible factors of degree up to certain value m (such polynomials are called m-rough). Gao and Panario [2] considered the case $m = O(\log n)$ and proved:

Theorem 5. *Denote by $P_q(n,m)$ the probability of a random monic polynomial of degree n over $\mathbb{F}_q$ being m-rough. Then when $n \to \infty$ and uniformly for q and $1 \le m \le O(\log n)$,*

$$P_q(n,m) = \prod_{k=1}^{m} \left(1 - \frac{1}{q^k}\right)^{I_k} (1 + o(1)),$$

Theorem 6. *Let $g_q(m) = \prod_{k=1}^{m} \left(1 - \frac{1}{q^k}\right)^{I_k}$. Then, for any prime power q and positive integer m,*

$$e^{-H_m} \le g_q(m) \le \left(1 - \frac{1}{\sqrt{q}}\right)^{-\frac{q}{q-1}} e^{-H_m}.$$

When $q \to \infty$, we have

$$g_q(m) = \prod_{k=1}^{m} \left(1 - \frac{1}{q^k}\right)^{I_k} \to e^{-H_m} \sim \frac{e^{-\gamma}}{m},$$

where γ is Euler's constant and $e^{-\gamma} = 0.56416\ldots$.

5. Discrete logarithm problem

For any element $b \in \mathbb{F}_q$, $b \neq 0$, there exists an integer x, $0 \le x \le q-2$, such that $b = \alpha^x$, where α is a generator. We call x the *discrete logarithm* of b in the base α.

We present here the *index calculus algorithm* to compute the discrete logarithm of any $b \in \mathbb{F}_q$, $b \neq 0$ and restrict ourselves to the case of $\mathbb{F}_{2^n}$.

This method consists of two parts. First, one builds a large database of logarithms by finding the logarithms of all irreducible polynomials of degree at most m, where m is a fixed positive integer. Second, one computes individual logarithms. To compute the logarithm of an element $g \in \mathbb{F}_{2^n}$, $g \neq 0$, one takes a random integer a, computes $h = g \cdot \alpha^a$, where α generates $\mathbb{F}_{2^n}$ and factors h in $h = \prod_{i=1}^{t} p_i^{e_i}$. If each irreducible factor p_i has degree $p_i \leq m$, then

$$\log g = \sum_{i=1}^{t} e_i \log p_i - a,$$

which can be easily evaluated by looking up in the database. If not all p_i have degree $\leq m$, then one generates another integer a and repeats.

Theorem 7 ([4]). *Let $\mathbb{F}_q$ be fixed, $f_m(z) = \prod_{k=1}^{m}(1-z^k)^{-I_k}$, and $r_0 = r_0(n,m)$ be the unique solution in $(0,1)$ of the equation $r_0(f_m'/f_m)(r_0) = n$, and let*

$$b(r_0) = \left(\frac{f_m{}'}{f_m}(r)\right)'\Bigg|_{r=r_0}.$$

Then, for

$$(\log n)(\log\log n)^{-1} \leq m \leq n \log\log n (\log n)^{-1}, \qquad n \to \infty,$$

$$[z^n] f_m(z) = (1+o(1)) \frac{f_m(r_0) r_0^{-n}}{\sqrt{(2\pi b(r_0)}}.$$

Soundararajan (1995) completed the full range of m estimating $[z^n]f_m(z)$ using recurrences relations. This could be done using partial fraction expansions for $1 \leq m \leq (\log n)(\log\log n)^{-1}$, and singularity analysis for $n \log\log n (\log n)^{-1}$.

6. Conclusions

Generating functions and singularity analysis allow for counting random polynomials over finite fields. We applied this methodology to give precise average-case analysis of a complete polynomial factorization algorithm [1]. Using this methodology, von zur Gathen, Gourdon & Panario (work in progress) present further research related to the average-case analysis of polynomial factorization algorithms. This work centers around [3], and the factoring algorithms of the 90's.

We also commented on other problems using polynomials over finite fields: tests and constructions of irreducible polynomials [2]; discrete log in $\mathbb{F}_{2^n}$ [4]; (see also Panario & Viola, work in progress).

References

[1] Flajolet (Philippe), Gourdon (Xavier), and Panario (Daniel). – *Random Polynomials and Polynomial Factorization.* – Research Report n° 2852, Institut National de Recherche en Informatique et en Automatique, March 1996. To appear in the *Proceedings of ICALP'96, Lecture Notes in Computer Science*.

[2] Gao (Shuhong) and Panario (Daniel). – Density of normal elements. *Finite Fields and their Applications*, vol. 3, n° 2, 1997, pp. 141–150.

[3] Kaltofen (E.) and Shoup (V.). – Subquadratic-time factoring of polynomials over finite fields. *Mathematics of Computation*, April 1998. – To appear.

[4] Odlyzko (A. M.). – Discrete logarithms in finite fields and their cryptographic significance. In *Advances in cryptology. Lecture Notes in Computer Science*, vol. 209, pp. 224–314. – Berlin, 1985. Proceedings of a conference held in Paris, 1984.

[5] Rabin (Michael O.). – Probabilistic algorithms in finite fields. *SIAM Journal on Computing*, vol. 9, n° 2, 1980, pp. 273–280.