12.1 Motivation

In this lecture, we comment on several problems in finite fields, their applications, and their relations with analytic combinatorics. Algebraic algorithms that deal with polynomials over finite fields can often be analyzed counting polynomials with particular properties.

We show that the most important characteristics of these algorithms can be treated systematically by the methodology of the course, that is, by establishing generating functions and studying their asymptotic analysis.

The main reference for the material of this lecture is the paper [7].

12.2 Schedule of this and the next lecture

We focus in this and the coming lecture on three problems over finite fields:

- polynomial factorization,
- irreducibility tests for polynomials,
- discrete logarithms.

For each problem,

- we present efficient algorithms,
- we derive interesting counting polynomials expressions, and
- we mention known results.

12.3 Arithmetic in finite fields

The polynomials we will consider have coefficients in $\mathbb{F}_q$. Our measure of complexity is the number of operations in $\mathbb{F}_q$. The algorithms we will present have operations such as multiplications, gcds, and divisions. The costs of these operations are as follows.

Two polynomials of degree $< n$ in $\mathbb{F}_q[x]$ can be multiplied with

- $O(n^2)$ operations in $\mathbb{F}_q$ using straight forward methods, or with
- $O(n \log n \log \log n)$ operations in $\mathbb{F}_q$ using FFT methods [17].

Division with remainder can be computed with the same costs. The gcd of two polynomials of degree $< n$ in $\mathbb{F}_q[x]$ can be computed with

- $O(n^2)$ operations in $\mathbb{F}_q$ using straight forward methods, or with
- $O(n \log^2 n \log\log n)$ operations in $\mathbb{F}_q$ using FFT methods [1].

For polynomials $f, g \in \mathbb{F}_q[x]$, $\deg f = n$, the computation of $g^q \bmod f$ can be done using the repeated squaring method. Example: repeated squaring for $g^{23} \bmod f$:

- Compute $g_1 = g \bmod f, g_2 = g_1^2 \bmod f$, $g_4 = g_2^2 \bmod f$, $g_8 = g_4^2 \bmod f, g_{16} = g_8^2 \bmod f$.
- Then, $g^{23} \bmod f = g_1 \cdot g_2 \cdot g_4 \cdot g_{16} \bmod f$.

In general, $O(\log q)$ multiplications in $\mathbb{F}_q[x]$ modulo f (later we will be more precise about the constant embedded in the big-Oh). That is, $O(n^2 \log q)$ operations in $\mathbb{F}_q$ using straight forward methods, or $O(n \log q \log n \log\log n)$ operations in $\mathbb{F}_q$ using FFT based methods. An excellent text that shows all these algorithms and the ones presented later in this and the coming lecture is [9].

We fix a finite field $\mathbb{F}_q$ with $q = p^m$ (p prime). All analyses are expressed as asymptotic forms in n, the degree of the considered polynomial.

As our interest is in *dominant asymptotics*, we restrict our attention to the costs of products, gcds and division with remainder that we assume to have constants τ_1, τ_2 and τ_3, respectively. These constants vary with the algorithm and machine used, but they are independent of n.

12.4 Generating functions

We focus our discussion on polynomials over a finite field $\mathbb{F}_q$. In the following, we review and extend some results already covered in previous lectures.

Let $\mathcal{I}$ be the collection of all monic irreducible polynomials in $\mathbb{F}_q$. Formally, the family of all monic squarefree polynomials and all monic polynomials are:

$$Q = \prod_{\omega \in \mathcal{I}} (1 + \omega), \qquad \text{and} \qquad \mathcal{P} = \prod_{\omega \in \mathcal{I}} (1 - \omega)^{-1}.$$

In this context, $\mathcal{I} = \sum_{\omega \in \mathcal{I}} \omega$.

Let z be a formal variable, and $|\omega|$ the degree of $\omega \in \mathcal{I}$. As we have already seen, the substitution $\omega \mapsto z^{|\omega|}$ produces generating functions. For instance,

$$I(z) = \sum_{\omega \in \mathcal{I}} z^{|\omega|} = \sum_n I_n z^n,$$

where I_n is the number of irreducible polynomials of degree n.

The same substitution applied to $\mathcal{P}$ and $\mathcal{Q}$ yields

$$Q(z) = \prod_{\omega \in \mathcal{I}} (1 + z^{|\omega|}) = \prod_{n=1}^{\infty} (1 + z^n)^{I_n},$$

and

$$P(z) = \prod_{\omega \in \mathcal{I}} (1 - z^{|\omega|})^{-1} = \prod_{n=1}^{\infty} (1 - z^n)^{-I_n}.$$

The coefficient $Q_n = [z^n]Q(z)$ equals the number of monic squarefree polynomials of degree n, and the coefficient $P_n = [z^n]P(z)$ represents the number of monic polynomials of degree n.

First we will determine I_n. Since P_n has value q^n, we have $P(z) = (1 - qz)^{-1}$, and the relations for $P(z)$ implicitly determine I_n:

$$P(z) = (1 - qz)^{-1} = \prod_{n=1}^{\infty} (1 - z^n)^{-I_n}.$$

Taking the logarithm of both sides and using $\log \frac{1}{1-z} = \sum_{j=1}^{\infty} \frac{z^j}{j}$ gives:

$$\log (1 - qz)^{-1} = \log \left(\prod_{n=1}^{\infty} (1 - z^n)^{-I_n} \right) = \sum_{k=1}^{\infty} I_k \log \frac{1}{1 - z^k} = \sum_{k=1}^{\infty} I_k \sum_{j=1}^{\infty} \frac{z^{jk}}{j}.$$

Interchanging sums and noting that $\sum_{k=1}^{\infty} I_k (z^j)^k = I(z^j)$ from above gives:

$$\log (1 - qz)^{-1} = \sum_{j=1}^{\infty} \frac{I(z^j)}{j}.$$

Since

$$[z^n] \log \frac{1}{1 - qz} = \frac{q^n}{n}$$

and

$$[z^n] \sum_{j=1}^{\infty} \frac{I(z^j)}{j} = \sum_{j|n} \frac{I_{n/j}}{j}$$

(where we want degree $n = jk$ so we want all possible j such that $j|n$), then

$$\frac{q^n}{n} = \sum_{j|n} \frac{I_{n/j}}{j}.$$

To solve for I_n in terms of q^n/n we need to use a classical tool from elementary number theory: the Moebius inversion formula.

Moebius inversion formula

If $f(n) = \sum_{d:d|n} g(d)$, then $g(n) = \sum_{d:d|n} \mu(d) f(n/d)$, where μ is the Moebius function:

$$\mu(d) = \begin{cases} 1 & \text{if } d = 1, \\ (-1)^k & \text{if } d \text{ is a product of } k \text{ distinct primes,} \\ 0 & \text{if } d \text{ contains repeated primes.} \end{cases}$$

In our case, the Moebius inversion formula gives

$$nI_n = \sum_{j|n} \mu(j) q^{n/j},$$

and since $\mu(1) = 1$, we have

$$I_n = \frac{q^n}{n} + O\left(\frac{q^{n/2}}{n}\right).$$

This fact about the behavior of I_n was known by several authors [2, 11].

Since $q \geq 2$ then $q^n \gg q^{n/2}$ in most situations, and q^n/n is the asymptotic result. The probability of obtaining an irreducible polynomial by uniform sampling from $\mathbb{F}_q$ is thus

$$\frac{I_n}{q^n} = \frac{1}{n} + O\left(\frac{1}{q^{n/2}}\right).$$

Therefore, a fraction very close to $1/n$ of the polynomials of degree n over $\mathbb{F}_q$ are irreducible, and in n samples, we would expect to find one irreducible polynomial. This information is useful when searching for irreducible polynomials (a topic we will cover in next lecture).

To determine Q_n, we use the Euler identity $(1-z)^{-1} = (1+z)(1+z^2)(1+z^4)\cdots$ applied to the infinite products for $P(z), Q(z)$ which entails

$$P(z) = Q(z) \cdot Q(z^2) \cdot Q(z^4) \cdots .$$

Thus,

$$Q(z) = \frac{P(z)}{P(z^2)} = \frac{1-qz^2}{1-qz},$$

and

$$Q_n = q^{n-1}(q-1)$$

for $n \geq 2$, with $Q_0 = 1$, and $Q_1 = q$. This was first noticed by Carlitz [4].

12.5 Parameters

Let χ be some integer–valued parameter on the collection $\mathcal{P}$ of monic polynomials. Consider

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

The coefficient $[z^n u^k]A(z,u)$ represents the number of polynomials of degree n and 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, for the average we have

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

12.6 Asymptotic analysis

As we have seen in this course, generating functions encode exact information in their coefficients. Their behavior near their dominant singularity is an important source of coefficient asymptotics.

Most of the generating functions $f(z)$ of interest here are singular at $z = 1/q$ with an isolated singularity of algebraic-logarithmic type. In that case, an expansion near $z = 1/q$ of the form

$$f(z) = \frac{1}{(1-qz)^{\alpha}} \left(\log \frac{1}{1-qz}\right)^{k} (1 + o(1))$$

can be translated to coefficients

$$[z^n]f(z) = q^n \frac{n^{\alpha-1}}{\Gamma(\alpha)} (\log n)^k (1 + o(1)),$$

whenever $\alpha \neq 0, -1, -2, \ldots$, by the method known as singularity analysis due to Flajolet & Odlyzko [8]. This requires analytic continuation (isolated singularity).

The same translation can be effected under a variety of alternative conditions corresponding either to Darboux method or to Tauberian theorems (for instance, see [16]).

12.7 Factorization of polynomials

Let us start defining the problem. Let $\mathbb{F}_q$ be a finite field with q elements, where q is a power of a prime.

> Given a monic univariate polynomial $f \in \mathbb{F}_q[x]$, find the complete factorization $f = f_1^{e_1} \cdots f_r^{e_r}$, where the f_i's are monic distinct irreducible polynomials and $e_i > 0$, $1 \leq i \leq r$.

A survey with algorithms and many more information related to this problem is in [10].

12.8 Applications

- Algebraic coding theory [2];
- Computer algebra [6, 9, 12, 13];
- Cryptography [5, 14, 15];
- Computational number theory [3].

Chor and Rivest public key cryptosystem [5]

Let $q = p^m$, $m \geq 2$, and $\mathbb{F}_q$ represented by an irreducible polynomial $f \in \mathbb{F}_p[x]$ of degree m. Let $g \in \mathbb{F}_p[x]$ be a primitive element in $\mathbb{F}_q$.

For each $0 \leq i < p$ determine a_i, with $0 \leq a_i < p^m - 1$, such that

$$x - i \equiv g^{a_i} \bmod f.$$

Finally, let Π be a permutation on $\{0, 1, \ldots, p-1\}$ such that $b_i = a_{\Pi(i)}$, for $i = 0, \ldots, p-1$.

Messages: $M = (M_0, \ldots, M_{p-1})$, $M_i \geq 0$ integers such that $\sum_{i=0}^{p-1} M_i < m$.

Public key: $\{b_0, b_1, \ldots, b_{p-1}\}$.

Private key: f, g and Π.

Encryption: For message M compute

$$E(M) = \sum_{i=0}^{p-1} b_i M_i \bmod (q-1).$$

Decryption: For cyphertext $c = E(M)$ compute

$$t \equiv g^c \equiv \prod_{i=0}^{p-1} (x - \Pi(i))^{M_i} \bmod f.$$

Since $\sum_{i=0}^{p-1} M_i < m$,

$$t(x) = \prod_{i=0}^{p-1} (x - \Pi(i))^{M_i}.$$

Factor t, and with the multiplicities of the factors obtain the message M.

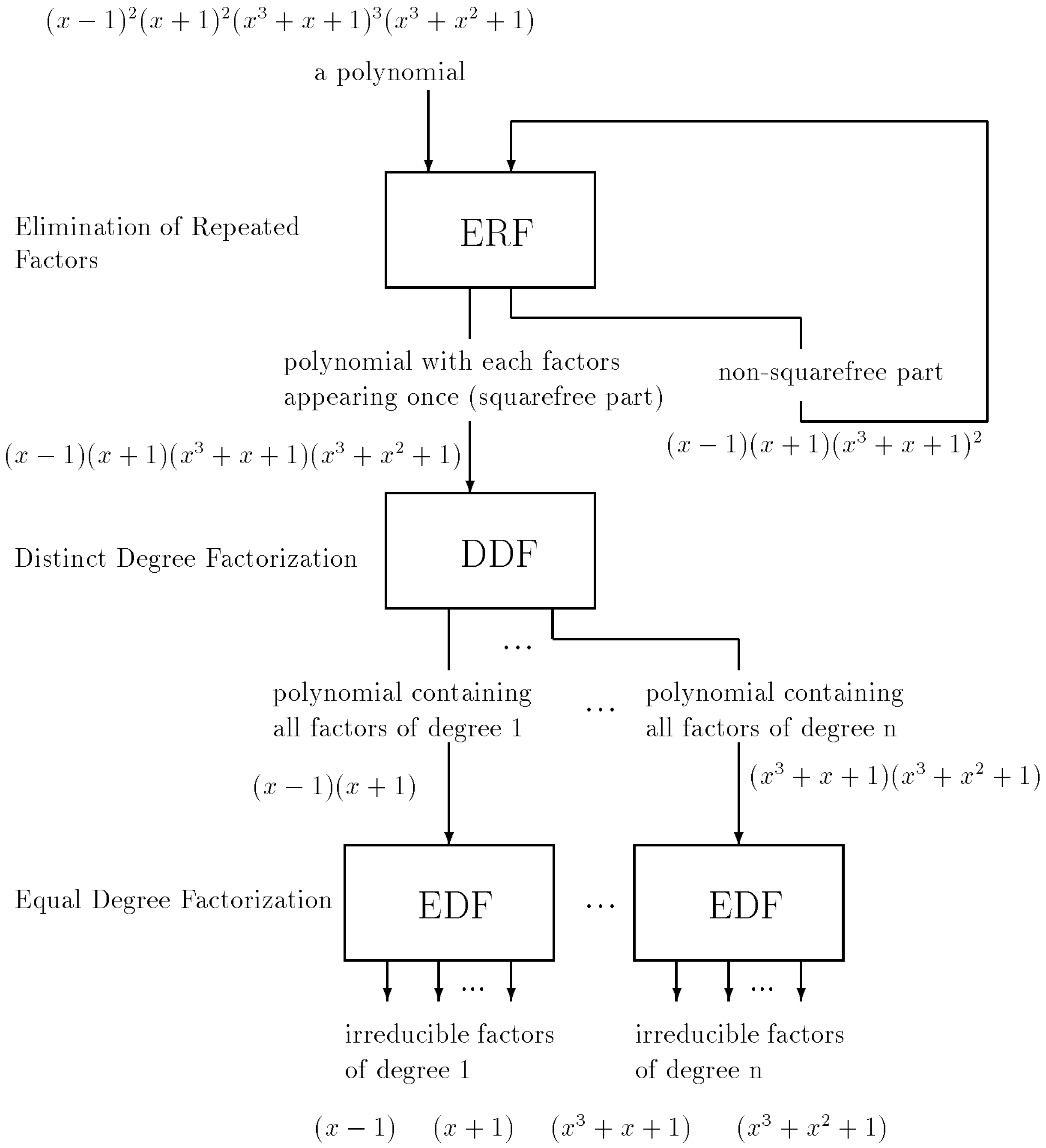
12.9 A general factorization algorithm

ERF *Elimination of repeated factors* replaces a polynomial by a squarefree one which contains all the irreducible factors of the original polynomial with exponents reduced to one.

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

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

Example:



Note: the polynomial on the input is not in the factored form of the picture but in extended form, that is: $x^{16}+x^{15}+x^{14}+5\,x^{13}+x^{12}-x^{11}+4\,x^{10}-7\,x^9-8\,x^8-7\,x^6-3\,x^5+5\,x^4+2\,x^3+2\,x^2+3\,x+1$.

```
procedure factor(f : polynomial);
     1: a := ERF(f);
     2: b := DDF(a);
        F := 1;
     3: for k from 1 to n do
             F := F.EDF(b[k],k);
        od;
     4: c := factor(f/a);
return(F.c);
end;
```

12.10 Elimination of repeated factors (ERF)

The first stage in the general factorization procedure is to produce a horizontal cut in the polynomial that leaves only one presence of each irreducible factor. The key idea of the elimination of repeated factors process is to compute $\gcd(f, f')$, where f is the polynomial to be factored. When working over a finite field, one has to consider the special case when a factor appears repeated p times, where p is the characteristic of the field, that is $q = p^m$. This complicates a bit the presentation of the algorithm though not its difficulty (see [9, 10]).

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

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

$$N_q = \sum_{k \geq 1} \frac{kI_k}{q^{2k} - q^k}.$$

When $q \to \infty$, $N_q \sim 1/q$.

The bivariate generating function that corresponds to the degree of the non-squarefree part of monic polynomials in $\mathbb{F}_q[x]$ is

$$P(z, u) = \prod_{n \geq 1} \left(1 + \frac{z^n}{1 - u^n z^n}\right)^{I_n}.$$

The mean degree of the non-squarefree part is obtained from the derivative $P_u(z, u)$ by singularity analysis.

The asymptotic value of N_q as $q \to \infty$ is obtained by means of the expansion

$$n\, I_n = q^n + O(q^{n/2}).$$

Since $n\,I_n = q^n + O(q^{n/2})$, then $nI_n \sim q^n$ for large n. Therefore,

$$N_q = \sum_{k\geq 1} \frac{kI_k}{q^{2k}-q^k} \sim \sum_{k\geq 1} \frac{q^k}{q^{2k}-q^k} = \sum_{k\geq 1} \frac{1}{q^k-1} \approx \sum_{k\geq 1} \frac{1}{q^k} = \frac{1}{1-\frac{1}{q}} - 1 = \frac{1}{q-1} \approx \frac{1}{q}.$$

Thus, the degree of the part that comes back is constant and the procedure rarely recurses. When it does, the piece that comes back is small and it doesn't matter too much how you do ERF because most of the time the polynomial is squarefree.

Consequences of Theorem 1 for the recursive structure of `factor`: the overall cost of the recursive calls remains $O(1)$ on average; alternative strategies giving the full squarefree factorization will lead to asymptotically equivalent costs; and 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*

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

References

[1] A.V. Aho, J.E. Hopcroft, and J.D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, Reading MA, 1974.

[2] E.R. Berlekamp. *Algebraic coding theory.* McGraw Hill, New York NY, 1968.

[3] J. Buchmann. Complexity of algorithms in algebraic number theory. In *Number Theory. Proc. First Conf. Canadian Number Theory Assoc.*, pages 37–53. Walter de Gruyter, 1990.

[4] L. Carlitz. The arithmetic of polynomials in a Galois field. *Amer. J. Math.*, 54:39–50, 1932.

[5] B. Chor and R. Rivest. A knapsack-type public key cryptosystem based on arithmetic in finite field. *IEEE Trans. Inform. Theory.*, 34:901–909, 1988.

[6] G.E. Collins. Factoring univariate integral polynomials in polynomial average time. In *Proc. EUROSAM 79*, volume 72 of *Lecture Notes in Computer Science*, pages 317–329, 1979.

[7] P. Flajolet, X. Gourdon, and D. Panario. The complete analysis of a polynomial factorization algorithm over finite fields. Submitted March 1998. [Extended abstract in *Proc. 23rd ICALP Symp.*, Lecture Notes in Computer Science, vol. 1099, p. 232–243, 1996.] Full version in technical report 3370, INRIA, March 1998.

[8] P. Flajolet and A. Odlyzko. Singularity analysis of generating functions. *SIAM Journal on Discrete Mathematics 3*, 2:216–240, 1990.

[9] J. von zur Gathen and J. Gerhard. *Modern computer algebra.* Cambridge University Press, 1998. In preparation.

[10] J. von zur Gathen and D. Panario. A survey on factoring polynomials over finite fields. To appear in the special issue of the MAGMA conference in *J. Symb. Comp.*, 1999.

[11] C.F. Gauss. *Untersuchungen úber Höhere Mathematik.* Chelsea, New York, 1889.

[12] K.O. Geddes, S.R. Czapor, and G. Labahn. *Algorithms for Computer Algebra.* Kluwer Academic Publishers, Boston, 1992.

[13] D.E. Knuth. *The art of computer programming, vol.2: seminumerical algorithms.* Addison-Wesley, Reading MA, 3 edition, 1997.

[14] H.W. Lenstra. On the Chor-Rivest knapsack cryptosystem. *J. of Cryptology*, 3:149–155, 1991.

[15] A. Odlyzko. Discrete logarithms and their cryptographic significance. In *Advances in Cryptology, Proceedings of Eurocrypt 1984*, volume 209 of *Lecture Notes in Computer Science*, pages 224–314. Springer-Verlag, 1985.

[16] A. Odlyzko. Asymptotic enumeration methods. In R. Graham, M. Grötschel, and L. Lovász, editors, *Handbook of Combinatorics*, volume 2, pages 1063–1229. Elsevier, 1995.

[17] A. Schönhage and V. Strassen. Schnelle Multiplikation großer Zahlen. *Computing*, 7:281–292, 1971.