CS681 **Computational Number Theory**

Lecture 11: Cantor-Zassenhaus Algorithm

Instructor: Piyush P Kurur *Scribe: Ramprasad Saptharishi*

Overview

In this class, we shall look at the Cantor-Zassenhaus randomized algorithm for factoring polynomials over $\mathbb{F}_p$. We shall do it for the case when $p \neq 2$. The case when $p = 2$, which isn't too different from the other case, would be given as an exercise for the students to solve.

1 Irreducibility Testing

We left off last class with a hint that the distinct degree factorization infact gives a straightforward irreducibility test. Here is the explicit answer.

We are given an f and we need to check if this polynomial is irreducible or not. First check if it is square free. If it isn't, immediately reject it. Else, proceed to compute the distinct degree factors of f. If the degree of f is n, the DDF algorithm returns $g_1, g_2, \cdots, g_n$ such that each g_i is the product of irreducible factors of f of degree d.

Now, suppose f was irreducible, then clearly every $g_i = 1$ for $1 \leq i \leq n-1$ and $g_n = f$. Thus just check if the returned g_i's satisfy this condition.

1.1 Generating Irreducible Elements

Suppose are given a positive integer d, we want to efficiently find an irreducible polynomial of degree d over $\mathbb{F}_p$. Now before we get into this, why is this important?

The answer is that this is the only way we can construct the field $\mathbb{F}_{p^d}$. There are lots of applications where we need to do arithmetic over a field of large size and $\mathbb{F}_p$ would be a candidate only if the prime p is large. And finding such a large prime is hard and inefficient.

Instead, we pick a small prime p and try and find an irreducible polynomial of degree d. Once we do that, we have $\mathbb{F}_p[X]/(f(X))$ which is isomorphic to $\mathbb{F}_{p^d}$. This is precisely finding irreducible polynomials of a given degree is very useful.

To generate an irreducible polynomial of degree d, we shall just randomly pick a polynomial of degree d. It can be argued that the probability that this polynomial is irreducible is pretty high. And since we even have a deterministic test to check if a polynomial f is irreducible, we just repeat this procedure: Pick an $f \in \mathbb{F}_p[X]$ of degree d at random and repeat this if the irreducibility test says that this polynomial is not irreducible.

All we need to do is to argue that the density of irreducible polynomials is large.

Theorem 1. *Let $I(d)$ be the number of irreducible polynomials of degree d over $\mathbb{F}_p$. Then*

$$I(d) = \frac{p^d}{d} + O\left(\sqrt{p}\right)$$

And therefore,

$$\Pr_{f \in \mathbb{F}_p[X], deg(f)=d}[f \in \mathrm{Irr}(\mathbb{F}_p, d)] \geq \frac{\frac{p^d}{d} + O(\sqrt{p})}{p^d} \geq \frac{1}{d}$$

As for the proof of the theorem, here is a sketch of it.

Proof. (sketch) We know that

$$X^{p^m} - X = \prod_{\substack{f \in \mathrm{Irr}(\mathbb{F}_p, d) \\ d|m}} f(X)$$

Comparing the degrees on both sides,

$$p^m = \sum_{d|m} I(d) \cdot d$$

Equations of this kind can be inverted using the *Möbius Inversion.*

Lemma 2 (Möbius Inversion)**.** *If we have any equation of the form*

$$f(m) = \sum_{d|m} g(d)$$

then

$$g(m) = \sum_{d|m} \mu(d) f(m/d)$$

Exercise: Read up on the Möbius Inversion.

With the inversion formula, once can complete the proof of the theorem by taking $f(m) = p^m$ and $g(m) = I(m) \cdot m$. $\square$

2 The Cantor-Zassenhaus Algorithm

Now we get to factoring a polynomial over $\mathbb{F}_p$. Given a polynomial of degree f over $\mathbb{F}_p$, it is enough to get one non-trivial[1] factor of f.

As we said in the last few lectures, the first thing to do is to check if f is square free. If it isn't we can just return the square-free part of f as a factor and be done. If it is square-free, we compute the distinct degree factorization of f. If f turns out to be irreducible, we just return "irreducible." Else, we have to proceed to find a non-trivial factor of f.

We shall factor each g_i returned by the DDF algorithm separately. Hence, we now assume that we have an $f \in \mathbb{F}_p[X] = g_1 \cdots g_m$ such that each g_i is irreducible, distinct and of the same degree d.

Here enters our old friend Chinese Remaindering. Since $f = g_1 \cdots g_m$, we know that

$$\mathbb{F}_p[X]/(f(X)) \cong \mathbb{F}_p[X]/(g_1(X)) \times \cdots \times \mathbb{F}_p[X]/(g_m(X))$$

Now note that each g_i is an irreducible polynomial of degree d. And therefore, $\mathbb{F}_p[X]/(g_i(X))$ is isomorphic to $\mathbb{F}_{p^d}$. Hence the product just looks like

$$\mathbb{F}_p[X]/(f(X)) \cong \mathbb{F}_{p^d} \times \cdots \times \mathbb{F}_{p^d}$$

And further, we know that

$$(\mathbb{F}_p[X]/(f(X)))^\star \cong \mathbb{F}_{p^d}^\star \times \cdots \mathbb{F}_{p^d}^\star$$

Now what do zero divisors, say g, in $\mathbb{F}_p[X]/(f)$ look like? When you take the image under the chinese remaindering, it should go to some tuple $(a_1, a_2, \cdots, a_m)$ where some $a_i = 0$. Further, if this zero divisor is non-trivial (0 is a trivial zero-divisor, useless), some other $a_j \neq 0$. What does this mean? g has a 0 in coordinate i which means that g is divisible by g_i, and hence $g \neq 1$. And also, g is non-zero at coordinate j and therefore g_j does not divide g and hence $g \neq f$. Thus, $gcd(g, f)$ is certainly not f nor 1 and hence is a non-trivial factor of f.

[1] trivial factors of f are 1 and f. Factors though they may be, are useless for us.

Therefore, the problem of finding factors reduces to the problem of finding zero divisors in $\mathbb{F}_p[X]/(f(X))$.

2.1 Finding Zero-Divisors

The idea is the following. We cross our fingers and pick a polynomial $a(X)$ of degree less than n at random. This is some element from $\mathbb{F}_p[X]/(f(X))$. If we are extremely lucky we might just get $gcd(a, f) \neq 1$, and this already gives us a non-trivial factor of f and we are done. Hence, lets assume that a is not a zero-divisor of the ring. And therefor, a must be an element of $(\mathbb{F}_p[X]/(f))^\star$, an invertible element.

Note that since we do not know the factors g_i, we do not know the chinese remainder map. We just know that a map exists, we don't know how to compute it. But suppose someone secretly told us that one of the coordinates of a under the chinese remainder map is -1, then what can we do?

Look at the images of $a(X) + 1$. The images of this is just 1 added to every coordinate of the image of $a(X)$. And since someone told us that one of the coordinates was -1, that coordinate in the image of $a(X)+1$ must be zero! Which means that, $a(X) + 1$ is a zero divisor. However it is possible that all the coordinates is -1 and that would just make $a(X)+1 = 0$ which is useless.

Now, how do we make sure that there is some -1 in one of the coordinates and not everywhere? Use the fact that each element of the product is $\mathbb{F}_{p^d}$. We know that $\mathbb{F}_{p^d}^\star$ is an abelian group of order $p^d - 1$. And therefore, for every element b in this group, $b^{p^d-1} = 1$.

We need a -1, and therefore we look at the square-root of it. Since we know that $b^{p^d-1} = 1$, $b^{(p^d-1)/2} = \sqrt{1}$ which can either be 1 or -1.[2] Let us just call $(p^d - 1)/2 = M$.

Thus, we have a simple procedure. Pick up some random polynomial $a(X)$ of degree less than n. Check if you are lucky by computing $gcd(a, f)$ and checking if it is 1. Else, compute $a(X)^{(p^d-1)/2} \bmod f(X)$ using repeated squaring. Now if a was mapping to $(a_1, a_2, \cdots, a_m)$, then a^M would be mapped to $(a_1^M, \cdots, a_m^M)$. And we just saw that each of a_i^M is either 1 or -1.

Claim 3. *Each $a_i^M = 1$ with probability $1/2$, and they are independent.*

[2] there can't be any other square-roots of 1. This is because square roots of -1 satisfy the equation $X^2 - 1 = 0$ and this equation can have only 2 roots in a field

Proof. Since the chinese remainder map is an isomorphism and each g_i's are distinct, they are clearly independent. To check that the probability that $b^M = 1$ is $1/2$, we look at the following map.

$$\begin{array}{rcl} \psi : \mathbb{F}_{p^d}^{\star} & \longrightarrow & \{1, -1\} \\ b & \longrightarrow & b^M \end{array}$$

Note that the set $\{1, -1\}$ form a group under multiplication. Infact it can be identified with the group $\mathbb{Z}/2\mathbb{Z}$.

Exercise: Prove that the map ψ is indeed a group homomorphism.

And therefore, the kernel of this map ψ is a subgroup of $\mathbb{F}_{p^d}^{\star}$ of all elements b such that $b^m = 1$. The other coset of this kernel is the set of elements b such that $b^M = -1$. Since these two are cosets, they are of equal size. Hence a randomly chosen b will have $b^M = 1$ with probability $1/2$. □

And therefore, each a_i is 1 or -1 with probability $1/2$. Thus the probability that all the coordinates are 1 or all the coordinates are -1 is just $1/2^m$. Thus with probability atleast $1 - 2^{m-1}$, we have some vector that has 1s at certain places and -1s at the rest. Thus, now we are in the case when someone had secretly told us that some coordinate is -1.

And therefore, we can pick a random polynomial $a(X)$, raise it to the power M modulo f, and add 1 to it. With probability atleast $1 - 2^{m-1}$, this will be a zero-divisor and hence $gcd(a^M + 1, f)$ will be a non-trivial factor of f.

So here is the algorithm:

Algorithm 1 Cantor-Zassenhaus Algorithm for Factoring

```
Input: A polynomial $f \in \mathbb{F}_p[X]$ of degree $n$.
if $f$ is not square-free then
    return the square-free part
end if
Compute the distinct degree factors of $f$. Call them $\{g_1, g_2, \cdots, g_n\}$.
if $g_n \neq 1$ then
    return IRREDUCIBLE
end if
for all $g_i \neq 1$ do
    EqualDegreeFactorize($g_i, d$)
end for
```

Algorithm 2 EQUALDEGREEFACTORIZE

```
Input: A polynomial $f \in \mathbb{F}_p[X]$ of degree $n$ all of whose $m$ irreducible factors are of degree $d$.
Pick a random polynomial $a(X)$ of degree less than $n$.
if $gcd(a, f) \neq 1$ then
    return $gcd(a, f)$
end if
Let $M = (p^d - 1)/2$.
Using repeated squaring, compute $a'(X) = a(X)^M + 1$.
if $a' \neq 0$ and $gcd(a', f) \neq 1$ then
    return $gcd(a', f)$ {Happens with probability atleast $1 - 2^{m-1}$}
end if
Repeat algorithm with a different choice for $a$.
```