6.885 Algebra and Computation October 3, 2005

Lecture 7

Lecturer: Madhu Sudan *Scribe: Jingbin Yin*

Today we are going to continue our talk about factorization of polynomials over finite fields. And then we will give a completely different deterministic algorithm to factorize polynomials over fields of small character.

1 Factorization of polynomials over finite field $\mathbb{F}_q$

1.1 Inspiration

In Lecture 5, we came up with a algorithm to find all linear factors of $f(x) \in \mathbb{F}_q[x]$, where q is odd:

LINEAR-FACTOR-FIND($f(x) = \sum c_i x^i$):

1. $f(x) \leftarrow \gcd(x^q - x, f(x))$.
2. If $\deg(f) = 0$, return Non. If $\deg(f) = 1$, return $f(x)$.
3. Pick $c \in \mathbb{Z}_q^*, d \in \mathbb{Z}_q$ at random.
4. $\tilde{f}(x) \leftarrow f(\frac{x-d}{c})$.
5. If $\tilde{h}(x) = \gcd(\tilde{f}(x), x^{\frac{q-1}{2}} - 1)$ is non-trivial,
   return LINEAR-FACTOR-FIND($h(x) = \tilde{h}(cx + d)$), LINEAR-FACTOR-FIND($\frac{f(x)}{h(x)}$).
   Else, try again from stage 3.

The efficiency of this algorithm based on three facts:

1. Every linear polynomial divides $x^q - x$.
2. We have a nice factorization of $x^q - x = x(x^{\frac{q-1}{2}} - 1)(x^{\frac{q-1}{2}} + 1)$.[1]
3. We have a efficient technique to randomize the factors of $f(x)$.

When we try to apply similar algorithm to find all factors of higher degree, it seems like impossible to randomize all the factors, since they may have different degrees. However, we could change the algorithm LINEAR-FACTOR-FIND a little bit, so that it may look like easier to generalize:

LINEAR-FACTOR-FIND-1($f(x) = \sum c_i x^i$):

1. $f(x) \leftarrow \gcd(x^q - x, f(x))$.
2. If $\deg(f) = 0$, return Non. If $\deg(f) = 1$, return $f(x)$.
3. Pick $c \in \mathbb{Z}_q^*, d \in \mathbb{Z}_q$ at random.
4. $h(x) = \gcd(f(x), (cx + d)^{\frac{q-1}{2}} - 1)$.

[1] This factorization is really nice because the degrees of the factors (except x) are all the same.

5. If $0 < \deg(h) < \deg(f)$,

   return LINEAR-FACTOR-FIND-1($h(x)$), LINEAR-FACTOR-FIND-1($\frac{f(x)}{h(x)}$).

   Else, try again from stage 3.

Then inspired by this algorithm, if we can apply similar algorithm to find all factors of higher degree in $\mathbb{F}_q[x]$, we have to deal with these three facts:

1. Every irreducible polynomial of degree d in $\mathbb{F}_q[x]$ divides $x^{q^d} - x$.
2. There is a nice factorization of $x^{q^d} - x = p_1(x)p_2(x)$ in $\mathbb{F}_q[x]$, i.e. $\deg(p_1) \approx \deg(p_2)$.
3. We can pick $g(x) \in \mathbb{F}_q[x]$ at random satisfying certain constraints, s.t. it is a good randomization.

If we have these three facts, we can try to find all irreducible factors of degree d of $f(x)$ by an algorithm similar to LINEAR-FACTOR-FIND-1.

Fact 1 is already proved in Lecture 6. We just focus on Fact 2 and 3 in the following.

1.2 Algorithm when q is odd

When q is odd, i.e. $q = p^k$, p is a odd prime, $k \in \mathbb{Z}$ and $k > 0$, we have $x^{q^d} - x = x(x^{\frac{q^d-1}{2}} - 1)(x^{\frac{q^d-1}{2}} + 1)$, which is a very nice factorization. Then we solved Fact 2. The only thing we have to do is to conduct Fact 3.

Suppose $f(x) = f_1(x)f_2(x)\cdots f_l(x)$, where $f_1(x), f_2(x), \cdots, f_l(x)$ are distinct irreducible polynomials of degree d, $l > 1$. What we need to do is to pick at random $g \in \mathbb{F}_q[x]$ satisfying some constraints, then try $h(x) = \gcd(f(x), g(x)^{\frac{q^d-1}{2}} - 1)$, s.t. the probability of $0 < \deg(h) < \deg(f)$ is as high as possible.

Consider $\mathbb{F}_q[x]/(f) = \mathbb{F}_q[x]/(f_1) \otimes \mathbb{F}_q[x]/(f_2) \otimes \cdots \otimes \mathbb{F}_q[x]/(f_l)$, where $\mathbb{F}_q[x]/(f)$ is a ring and $\mathbb{F}_q[x]/(f_i)$, $1 \le i \le l$, are fields. Let $g_i \equiv g \pmod{f_i}$, $1 \le i \le l$. The probability of $g_i^{\frac{q^d-1}{2}} - 1 = 0$(or $g_i^{\frac{q^d-1}{2}} - 1 \neq 0$)[2] in $\mathbb{F}_q[x]/(f_i)$ is roughly $\frac{1}{2}$, i.e. the probability of $f_i|h$ (or $f_i \nmid h$) is roughly $\frac{1}{2}$. Thus if we can choose at random $g_1 \in \mathbb{F}_q[x]/(f_1)$ and $g_2 \in \mathbb{F}_q[x]/(f_2)$ independently, the probability of $0 < \deg(h) < \deg(f)$ is roughly $1 - 2(\frac{1}{2})^2 = \frac{1}{2}$ (big enough). (Since $\deg(h) = 0$ only if $f_1 \nmid h$ and $f_2 \nmid h$, and $\deg(h) = deg(f)$ only if $f_1|h$ and $f_2|h$.) By Chinese Remainder Theorem, to choose at random $g_1 \in \mathbb{F}_q[x]/(f_1)$ and $g_2 \in \mathbb{F}_q[x]/(f_2)$ independently is equal to choose $g' \in \mathbb{F}_q[x]/(f_1f_2)$ at random. Thus we can complete Fact 3 as:

Pick $g \in \mathbb{F}_q[x]$ at random of $\deg(g) \le 2d - 1$.

By the three facts we completed, we can construct an algorithm to find all irreducible factors of degree d of $f(x) \in \mathbb{F}_q[x]$, if $f(x)$ is a product of irreducible polynomials of degree d:

[2] $\alpha \in \mathbb{F}_Q^*$ is called a **quadratic residue** if $\exists\beta \in \mathbb{F}_Q$, s.t. $\alpha = \beta^2$; $\alpha \in \mathbb{F}_Q^*$ is called a **quadratic non-residue** if $\nexists\beta \in \mathbb{F}_Q$, s.t. $\alpha = \beta^2$. Then α is a quadratic residue iff $\alpha^{\frac{Q-1}{2}} - 1 = 0$, and α is a quadratic non-residue iff $\alpha^{\frac{Q-1}{2}} - 1 \neq 0$.

d-SPLIT($f(x)$, where $f(x)$ is a product of irreducible polynomials of degree d)**:**

1. If $\deg(f) = 0$, return Non. If $\deg(f) = d$, return $f(x)$.
2. Pick $g \in \mathbb{F}_q[x]$ at random of $\deg(g) \leq 2d - 1$.
3. $h(x) = \gcd(f(x), g(x)^{\frac{q^d-1}{2}} - 1)$.
4. If $0 < \deg(h) < \deg(f)$,

   return d-SPLIT($h(x)$), d-SPLIT($\frac{f(x)}{h(x)}$).

   Else, try again from stage 2.

Finally from $d = 1$ to $d = \deg(f)$, we can construct an algorithm to find all factors of $f(x) \in \mathbb{F}_q[x]$:

FACTORIZE-ANY-POLY($f(x) \in \mathbb{F}_q[x]$**):**

1. If $\deg(f) = 0$ or 1, return f(x).
2. Repeat the following for $d = 1$ to $\deg(f)$.
3. Compute $f_d(x) = \gcd(f(x), x^{q^d} - x)$. ($f_d$ is the product of all distinct irreducible factors of degree d of $f(x)$.)
4. Find all irreducible factors $h_1, h_2, \cdots, h_k$ of f_d by d-SPLIT(f_d).
5. By removing all multiples of h_i in f, find $\alpha_i \in \mathbb{Z}, \alpha_i > 0$, s.t. $h_i^{\alpha_i} | f$ and $h_i^{\alpha^i+1} \nmid f$, $1 \leq i \leq k$.
6. return $h_1^{\alpha_1}, h_2^{\alpha_2}, \cdots, h_k^{\alpha_k}$.

1.3 Algorithm when $q = 2^t$

Now, the only problem we left after discussing in 1.2 is the condition when $\mathbb{F}_q$ is of the form $\mathbb{F}_{2^t}$, where $t \in \mathbb{Z}, t \geq 1$. First of all, we complete Fact 2 and 3.

For Fact 2, Let $p(x) = \Pi_{i \in \mathbb{F}_q}(T(x) - i)$, where $T(x)$ is the trace of $\mathbb{F}_{q^d}$ to $\mathbb{F}_q$, i.e. $T(x) = x + x^q + x^{q^2} + \cdots + x^{q^{d-1}}$. Since $\forall \alpha \in \mathbb{F}_{q^d}$, $T(\alpha) \in \mathbb{F}_q$, then we have $p(\alpha) = 0$ for all $\alpha \in \mathbb{F}_{q^d}$. Thus $p(x)$ is a polynomial of degree q^d which vanishes on $\mathbb{F}_{q^d}$, and hence must equal to $x^{q^d} - x$, the other polynomial of degree q^d that vanishes on $\mathbb{F}_{q^d}$. We conclude that $x^{q^d} - x = p(x) = \Pi_{i \in \mathbb{F}_q}(T(x) - i)$ in $\mathbb{F}_{q^d}[x]$, and hence in $\mathbb{F}_q[x]$. Then we can depart $\mathbb{F}_q$ to two parts A and B, s.t. $|A| = |B|$. Then let $p_1(x) = \Pi_{i \in A}(T(x) - i), p_2(x) = \Pi_{i \in B}(T(x) - i)$, we have $x^{q^d} - x = p_1(x)p_2(x)$ in $\mathbb{F}_q[x]$ and $\deg(p_1) = \deg(p_2)$. We complete Fact 2.

By similar discussing as in 1.2, we also can complete Fact 3 as:

Pick $g \in \mathbb{F}_q[x]$ at random of $\deg(g) \leq 2d - 1$.

Thus there is a similar algorithm as in 1.2, just replacing $g(x)^{\frac{q^d-1}{2}} - 1)$ by $p_1(g(x))$.

The algorithm's run time is a polynomial of $\deg(f)$ and $\log q$. That saved a lot of time when q is very large.

2 A deterministic algorithm

In this section, we want to talk about Berlekump's deterministic algorithm to factorize $f(x)$ over field $\mathbb{F}_q$ in time poly$(\deg(f), t, p)$, where $q = p^t$.

The essence of this algorithm is that:

CLAIM: 1. Given $f(x) \in \mathbb{F}_q[x]$ reducible, where $q = p^t$, there exists a polynomial $g \in \mathbb{F}_q[x]$ s.t. $f(x)|g(x)^p - g(x)$ and $0 < \deg(g) < \deg(f)$.
2. $g(x)$ can be found efficiently.

First of all, we discuss $g(x)$ for a moment to show why this claim is useful. Since $g(x)^p - g(x) = \Pi_{\alpha \in \mathbb{F}_p}(g(x) - \alpha)$, if we have $f(x)|g(x)^p - g(x)$, we can try $\gcd(f(x), g(x) - \alpha)$, $\forall \alpha \in \mathbb{F}_p$, there must be at least one of these a non-trivial factor of $f(x)$. By doing so repeatedly, we can factorize $f(x)$ completely.

Now, we have to prove the claim.

PROOF 1. Consider $f(x) = f_1(x)f_2(x)$, where $\gcd(f_1, f_2) = 1$ and $\deg(f_1), \deg(f_2) > 0$. Then $f(x)|g(x)^p - g(x)$ iff $f_1(x)|g(x)^p - g(x)$ and $f_2(x)|g(x)^p - g(x)$. By Chinese Remainder Theorem, there exists a polynomial $g(x) \in \mathbb{F}_q[x]/(f)$ s.t. $g(x) \equiv \alpha_1 (\mathrm{mod} f_1), g(x) \equiv \alpha_2 (\mathrm{mod} f_2)$, where $\alpha_1 \neq \alpha_2$ and $\alpha_1, \alpha_2 \in \mathbb{F}_p$. Thus $g(x)$ satisfies that $f(x)|g(x)^p - g(x)$ and $0 < \deg(g) < \deg(f)$.

2. (We just give a introduction to this part. The whole proof will be given on the next lecture.)
Suppose $g(x), h(x)$ both satisfy our needs. Then $g(x)^p - g(x) \equiv h(x)^p - h(x) \equiv 0 (\mathrm{mod} f)$. We have:

1. $(g(x) + h(x))^p - (g(x) + h(x)) \equiv 0 (\mathrm{mod} f)$.
2. $(\alpha g(x))^p - (\alpha g(x)) \equiv 0 (\mathrm{mod} f)$, $\forall \alpha \in \mathbb{F}_p$.

Thus, all the polynomial satisfying our needs form a vector space over $\mathbb{F}_p$.
Since we also can view $\mathbb{F}_q$ as a vector space over $\mathbb{F}_p$, let $g(x) = \sum_{i=0}^{\deg(f)-1} c_i x^i$, where $c_i \in \mathbb{F}_q = \mathbb{F}_p^t, c_i = (c_{i1}, c_{i2}, \cdots, c_{it})$. $g(x)$ is given by a vector of $t \deg(f)$ elements: $\{c_i j : 1 \leq i \leq deg(f), 1 \leq j \leq t\}$ in $\mathbb{F}_p$.
Then what we have to do is to find some linear constraints s.t. $g(x)$ satisfies our needs.
(To be continue.)