

Class Notes

Tuesday, September 5, 2006

Introduction

Given a ground field \mathbb{F}_q , if we want to build an extension of degree n we need an irreducible polynomial $f(x)$ of degree n so that $\mathbb{F}_{q^n} = \mathbb{F}_q[x]/(f(x))$. We shall now study some algorithms in $\mathbb{F}_q[x]$ with a view towards an irreducibility test.

Division Algorithm

Given polynomials $a(x), f(x) \in \mathbb{F}_q[x] \exists$ polynomials $b(x), r(x) \in \mathbb{F}_q[x]$ such that

1. $a(x) = b(x)f(x) + r(x)$
2. $\deg r(x) < \deg f(x)$

Algorithm

Let $a = a_0x^m + \dots$ and $f = f_0x^n + \dots$

If $\deg a < \deg f \Rightarrow b = 0, r = a$

Else if $\deg a = m > \deg f = n$

Replace a by $a - \frac{a_0}{f_0}x^{m-n}f$ and b by $b + \frac{a_0}{f_0}x^{m-n}$

Do this until $\deg a < \deg f$.

This algorithm takes at most $m - n$ steps to get b and r . Each step takes $O(n)$ operations in \mathbb{F}_q and the whole process takes $O(mn)$ operations.

Euclidean Algorithm

Given polynomials $a, b \in \mathbb{F}_q[x]$ with $\deg b \leq \deg a$ we want to compute $\gcd(a, b)$. We shall let $a \% f$ denote the remainder when a is divided by f .

Algorithm

$\gcd(a, b) = \gcd(b, a \% b)$

Do this until $\gcd(a, 0) = a$

Iterating will compute $\gcd(a, b)$ in $O(\max\{\deg a, \deg b\}) = O(\deg a)$ division of polynomials.

Raising to an integral power

Given $a, f \in \mathbb{F}_q[x]$ and $m > 0$ an integer we would like to compute $a^m \bmod f$. This can be done in $O(\log m)$ operations in $\mathbb{F}_q[x]/(f(x))$.

Algorithm

Let us look at the binary expansion of m .

$$m = m_0 + m_1 2 + \dots + m_r 2^r, m_i \in \{0, 1\}$$

To improve efficiency of our algorithm we could use the base p representation of m . We compute by squaring and then reducing $\text{mod } p$ the previous term of the sequence.

$$\{a, a^2, \dots, a^{2^r}\}$$

Then $a^m = a^{m_0} (a^2)^{m_1} \dots (a^{2^r})^{m_r}$ can be computed by using at most $r + 1$ multiplications from terms of the sequence.

Since we are reducing $\text{mod } f$ each time the degree of a does not become large.

Irreducibility of Polynomials

Theorem 1. *Let $f(x) \in \mathbb{F}_q[x]$ be a polynomial of degree n . Then $f(x)$ is irreducible iff*

1. $f(x) \mid (x^{q^n} - x)$
2. $\gcd(f(x), x^{q^d} - x) = 1, \forall d \mid n, d < n$

Proof. Assume 1 and 2.

1 $\Rightarrow f$ splits completely in \mathbb{F}_{q^n} and has simple roots.

2 $\Rightarrow f$ has no roots in a smaller subextension of $\mathbb{F}_{q^n}/\mathbb{F}_q$

So f is irreducible, since a factor would have to have roots in a field smaller than \mathbb{F}_{q^n} .

Converse is similar. □

Algorithm

We want to compute $(x^{q^d} - x) \% f$.

We compute x^{q^d} in $\mathbb{F}_q[x]/(f(x))$ which takes at most $d \log q$ steps.

$$x^{q^d} \equiv b \pmod{f} \quad (\deg b < \deg f)$$

$$(x^{q^d} - x) \% f = (b - x) \% f$$

When $d = n$ this is item 1 of the theorem. For $d < n$ this calculation is the first step of the Euclidean Algorithm. Subsequent steps involve only polynomials of degree at most n .

Example 1. Consider $f(x) = x^5 + x + 1$ in $\mathbb{F}_2[x]$.

Condition 2 of the Theorem is clearly satisfied i.e. $\gcd(f(x), x^2 - x) = 1$.

We want to compute $x^{2^5} \text{ mod } f$.

x, x^2, x^4

$$\begin{aligned} x^8 &\equiv x^4 + x^3 \pmod{f} \\ x^{16} &\equiv x^8 + x^6 \equiv x^4 + x^3 + x^2 + x \pmod{f} \\ x^{32} &\equiv x^8 + x^6 + x^4 + x^2 \\ &\equiv x^4 + x^3 + x^2 + x^4 + x^2 \equiv x^3 + x \pmod{f} \end{aligned}$$

Hence $(x^{32} - x) \% f = x^3 \neq 0$ and hence condition 1 of the Theorem is not satisfied. So f is reducible.

For large q and small n there might be better algorithms.