Comparing Families of Lattices for efficient Bounded Distance Decoding near Minkowski's Bound(Part 2)

Sasha

June 30, 2020

1 Lattice computation for polynomials

Let us set parameters a prime power q and integers k, d and n. Let $\mathbb{F}_q[x]$ be polynomial ring over a field $\mathbb{F}_q$. We take a set of k irreducible polynomials $c_j(x) \in \mathbb{F}_q[x]$, $j = 1, ..., k$ of degree d. According to the analogue of the prime number theorem for polynomials k must not be greater than $\frac{q^d}{d}$.

Define $c(x) \coloneqq \prod_{j=1}^{k} c_j(x)$. We are going to work in the multiplicative group of the quotient ring of $\mathbb{F}_q[x]$ with respect to $c(x)$. Chinese Remainder Theorem helps to determine the structure of $\big(\mathbb{F}_q[x]/c(x)\big)^*$:

$$\big(\mathbb{F}_q[x]/c(x)\big)^* \sim \prod_{i=1}^{k} \big(\mathbb{F}_q[x]/c_i(x)\big)^* \sim \prod_{i=1}^{k} \mathbb{F}_{q^d}^*$$

Multiplicative group of a field is cyclic, therefore, we can consider discrete logarithms in every component of the product to find a lattice basis.

Consider a vector $a = (\alpha_1, ..., \alpha_n) \in \mathbb{F}_q^n$ where α_is are pairwise different. Since polynomials $c_j(\cdot)$ are irreducible over $\mathbb{F}_q$ neither of α_i can be their root. So for all α_i we also have: $c(\alpha_i) \neq 0$.

Now consider a group morphism:

$$\psi : \mathbb{Z}^n \to \big(\mathbb{F}_q[x]/c(x)\big)^*$$

$$(u_1, ..., u_n) \mapsto \prod_{i=1}^{n} (x - \alpha_i)^{u_i} \pmod{c(x)}$$

Lattice is defined as the kernel of the morphism:

$$\mathcal{L} = \ker\psi = \{(u_1, ..., u_n) \in \mathbb{Z}^n | \prod_{i=1}^{n}(x - \alpha_i)^{u_i} \equiv 1 \pmod{c(x)}\}$$

Appying CRT gives us the following equivalence

$$\mathcal{L} = \ker\psi = \{(u_1, ..., u_n) \in \mathbb{Z}^n | \forall 1 \leq j \leq k : \prod_{i=1}^{n}(x - \alpha_i)^{u_i} \equiv 1 \pmod{c_j(x)}\}$$

Supposing we know β_j a generator of $\left(\mathbb{F}_q[x]/c_j(x)\right)^*$ for every j we get another representation:

$$\mathcal{L} = \{(u_1, ..., u_n) \in \mathbb{Z}^n | \forall 1 \leq j \leq k : \sum_{i=1}^{n} u_i log_{\beta_j}(x - \alpha_i) \equiv 0 \pmod{q^d - 1}\}$$

What might be confusing is that each $\log_{\beta_j}$ has a different input domain. For every j: $\log_{\beta_j}$ acts from $\left(\mathbb{F}_q[x]/c_j(x)\right)^*$ into $(\mathbb{Z}/(q^d - 1)\mathbb{Z})$.

We obtained a parity check representation of $\mathcal{L}$. To calculate a basis of $\mathcal{L}$ we can follow simplified version of the algorithm for integers. We obtain dual basis by scaling parity check matrix and concatenating it with I_n. Then we remove linear dependencies and finally obtain primal basis from the dual.

2 Building blocks

2.1 Factorization by trial division

Input: A polynomial g such that $deg(g) \leq m$ whose roots are among $\alpha_1, \ldots, \alpha_n \in \mathbb{F}_q$
Output: $e_1, \ldots, e_n$ s.t. $g = \prod_{i=1}^{n}(x - \alpha_i)^{e_i}$
There's only n possible roots, one trial division takes $O(m)$ time and the number of factors is bounded by m. So overall complexity is $O(m^2 n)$.

2.2 Rational function reconstruction

I found here an algorithm which they call Wang's algorithm for rational function reconstruction.
Goal: Given g, f find $n, d \in \mathbb{F}[x]$ that $deg(n) + deg(d) < \frac{deg(f)}{2}$ and $\frac{n}{d} = g \pmod{f}$
Algorithm:(function $lc()$ outputs the leading coefficient)

1. $r_0 = f$ $r_1 = g$
   $t_0 = 0$ $t_1 = 1$
   $q = 1$

2. While $deg(q) \leq \frac{deg(f)}{2}$ do
   $q = r_0 // r_1$
   $(r_0, r_1) = (r_1, r_0 - qr_1)$
   $(t_0, t_1) = (t_1, t_0 - qt_1)$

3. if $GCD(r_0, t_0) \neq 1$ or $deg(r_0) + deg(t_0) \geq \frac{deg(f)}{2}$:
   return FAIL
   else:
   return $(\frac{r_0}{lc(t_0)}, \frac{t_0}{lc(t_0)})$

Lemma 1. *Let $\mathbb{F}$ be a field, $f, g, r, s, t \in \mathbb{F}[x]$ with $r = sf + tg$, $t \neq 0$, $deg(f) > 0$, and $deg(r) + deg(t) < deg(f)$. Suppose r_i, s_i, t_i for $0 \leq i \leq l + 1$ be the elements of the ith iteration in the Extended Euclidean Algorithm for f and g (e.i. $r_i = s_i f + t_i g$).*
Then there exists a nonzero element $\alpha \in \mathbb{F}[x]$ such that $r = \alpha r_j$, $s = \alpha s_j$, $t = \alpha t_j$, where $deg(r_j) \leq deg(r) < deg(r_{j-1})$

Proof. (Lemma 3.2 page 35) □

So if the solution exists it must be one of the pairs (r_i, t_i) of the EEA.

ToDo: Add a lemma to prove the following: If $deg(n) + deg(d) \leq \frac{deg(f)}{2}$ than the solution corresponds to the unique row of the EEA where $deg(q) > \frac{deg(f)}{2}$

Question 1. *Check if we can use FEEA.*

2.3 Computing logs

To construct the lattice we need to compute the following: $\forall 1 \leq i, j \leq n$: $log_{\beta_j}(x - \alpha_i) \pmod{q^d - 1}$. The order of multiplicative group is $q^d - 1$ which might not be a smooth integer so we cannot use Pohlig-Hellman+Pollard pho. We can choose $q^d = n^{O(1)}$ $(poly(n))$ so group order is overall small(e.g. $d = constant$ and $q = n^{O(1)}$ does work, so does $q = constant$ and $d = O(\log n)$).

3 Decoding radius

3.1 Only positive discrete error

Suppose we receive $t = u + e$ where $u \in \mathcal{L}$, $\|e\|_1 \leq r_1$ and $\forall i : e_i \in \mathbb{N}$. Then we can compute

$$\prod_{i=1}^{n}(x-\alpha_i)^{t_i} = \prod_{i=1}^{n}(x-\alpha_i)^{u_i}\prod_{i=1}^{n}(x-\alpha_i)^{e_i} \pmod{c(x)}$$

If $\|e\|_1 = \sum_{i=1}^{n} e_i \leq deg(c) = d \cdot k$ the operation above will give us exactly the polynomial $\prod_{i=1}^{n}(x-\alpha_i)^{e_i}$. Then we can recover e_i, $1 \leq i \leq n$ from the factorization.
So $r_1 = d \cdot k$.

3.2 Arbitrary discrete error

Now we have $\forall i : e_i \in \mathbb{Z}$. Then

$$\prod_{i=1}^{n}(x-\alpha_i)^{t_i} \pmod{c(x)} = \prod_{i=1}^{n}(x-\alpha_i)^{e_i} = \frac{\prod_{i\in I}(x-\alpha_i)^{e_i}}{\prod_{j\in J}(x-\alpha_j)^{-e_j}}$$

Lemma 2. *Given g, c where $deg(c) = d \cdot k$ we can recover $f_1, f_2 \in \mathbb{F}[x]$ that $\forall i = 1; 2$: $deg(f_i) \leq \lfloor \frac{dk}{2} \rfloor$ and $\frac{f_1}{f_2} = g \pmod{c}$ in polynomial time.*

So we can decode every message for which $\|e\|_1 = \sum_{i=1}^{n} |e_i| \leq \lfloor \frac{dk}{2} \rfloor$

3.3 Normalized radius

Directly follows from [**?**]. $\bar{r}_1 = \frac{dk}{det(\mathcal{L})^{1/n}}$ where $det(\mathcal{L}) = \Phi(c(x)) = (q^d-1)^k$.

$$\bar{r}_1 = \frac{dk}{(q^d-1)^{k/n}}$$

What should be the values of d and k?

We have the following constraints: $q^d = n^{O(1)}$, $dk < q^d$, $n \leq q$ from here it follows that d must be constant

A note for myself! In my code the determinant of the lattice is changing! It is upper-bounded by $(q^d - 1)^k$ but practice shows that it is often much smaller.