A Polynomials

We briefly review some definitions and results related to polynomials. To avoid the confusion between X and χ, we follow the convention of using Z as the formal variable for polynomials.

A nonzero *polynomial* over a field $\mathcal{U}$ can be expressed by the following form

$$F(Z) = a_n Z^n + a_{n-1} Z^{n-1} + \cdots + a_0 \ (n \geq 0, a_n \neq 0)$$

where $a_0, a_1, \cdots, a_n \in \mathcal{U}$ and Z is a formal variable. The set of all polynomials over $\mathcal{U}$ is denoted by $\mathcal{U}[Z]$.

- The *degree* of a nonzero polynomial $F(Z)$ is defined as $\deg F := n$. For example, $\deg(Z^3 + 1) = 3$.
- If $a_n = 1$, we say $F(Z)$ is *monic*. For example, $Z^2 + 2Z$ is monic, while $3Z^2 + 4$ is not.
- $F(Z) \bmod G(Z)$ is defined as the remainder obtained when $F(Z)$ is divided by $G(Z)$. We say $F(Z) \equiv G(Z) \pmod{Z^n}$ if $F(Z) \bmod Z^n = G(Z) \bmod Z^n$.
- We say $F(Z)$ is divisible by $G(Z)$ if $F(Z) \bmod G(Z) = 0$, denoted as $G(Z) \mid F(Z)$. For example, we have $Z + 1 \mid Z^2 + 2Z + 1$.
- The $\gcd(F, G)$ is defined as the unique monic polynomial $H(Z)$ of the highest degree that both $F(Z), G(Z)$ are divisible by $H(Z)$. For example, $\gcd(Z^2 + 2Z + 1, Z^2 + 3Z + 2) = Z + 1$.
- If $\gcd(F, G) = 1$, we say that F and G are *coprime* to each other.

For polynomials $F(Z), G(Z), H(Z)$ with $\gcd(H(Z), G(Z)) = 1$, we define the result of *modular division* operation

$$\frac{F(Z)}{G(Z)} \bmod H(Z)$$

as the unique polynomial $R(Z)$ such that $\deg R < \deg H$ and

$$R(Z)G(Z) + H(Z)K(Z) = F(Z)$$

holds for some polynomial $K(Z)$. It can be shown that $R(Z)$ always exists uniquely.

The properties of polynomials under modular arithmetic are analogous to those of integers under modular arithmetic, so we will not elaborate further.

Naively multiplying two polynomials with degree n costs $\Theta(n^2)$ time, but using the *Number Theoretic Transform* (*NTT*)

We say $u \in \mathcal{U}$ is a *root* of $F(Z)$ if $F(u) = 0$. In our problem, $\mathcal{U}$ is a finite field, hence prior works represented by the Cantor–Zassenhaus algorithm

B Pseudocodes

In this part of the code, we have employed additional fingerprint verification. This part of the code is optional: if one desire minimizing communication overhead, simply remove it.

B.1 The Insertion of XYZ-v1

In the pseudo code, we maintain elements of the current group in S, rendering the algorithm an online algorithm. Here, $\times$ means multiplying two polynomials using NTT.

B.2 The Insertion of XYZ-v2

We can also use the insert operation of XYZ-v1 to substitute `InsertToCell`, but it is not necessary.

Algorithm 1: XYZ-v1 : Insertion Procedure

Function `CalcCharPoly(`T`)`**:**
  if $|T| = 1$ **then**
    Suppose $T = \{a\}$;
    return $Z - a$;
  Divide T into two parts L, R that $|L| = \left\lfloor \frac{|T|}{2} \right\rfloor, |R| = \left\lceil \frac{|T|}{2} \right\rceil$;
  return `CalcCharPoly(`L`)` $\times$ `CalcCharPoly(`R`)`;
Procedure `Insert(`x`)`**:**
  $sz_A \leftarrow (sz_A + 1) \bmod (2d + 1)$;
  $S \leftarrow S \cup \{x\}$;
  if $|S| = d$ *or this is the last insertion* **then**
    $\chi_A(Z) \leftarrow \chi_A(Z) \times$ `CalcCharPoly(`S`)` $\bmod Z^d$;
    $S \leftarrow \varnothing$;

Algorithm 2: XYZ-v2 : Insertion Procedure

Procedure `InsertToCell(`i, x`)`**:**
  $\mathcal{B}^c[i] \leftarrow (\mathcal{B}^c[i] + 1) \bmod (2l + 1)$;
  $\mathcal{B}^p[i] \leftarrow (\mathcal{B}^p[i] \times (Z - x)) \bmod Z^l$;
  $\mathcal{B}^{fp}[i] \leftarrow \mathcal{B}^{fp}[i] + h_{\text{fp}}(x)$;
Procedure `Insert(`x`)`**:**
  for $i \in [k]$ **do**
    `InsertToCell(`$h_i(x), x$`)`;

B.3 The Decoding of XYZ-v2

The overall decoding framework follows a similar structure to that of IBLT and related works, with modifications made to adapt the code for multivariate cells.

C Implementation of Modified RFR

We now proceed to prove the theorem stated in Section 3 and present our modifed RFR algorithm.

C.1 Uniqueness of Solution

As a beginning, we first prove the *uniqueness* of the solution.

Proof. We set $d_F := \lfloor \frac{n+m}{2} \rfloor, d_G := \lfloor \frac{n-m}{2} \rfloor$. One can see

$$\begin{aligned} &\deg F - \deg G = m, \deg F + \deg G \leq n \\ \implies &\deg F \leq d_F, \deg G \leq d_G \end{aligned}$$

If there are multiple solutions, we take (F_0, G_0) and (F_1, G_1) for example. We have

$$\begin{aligned} &\frac{F_0(Z)}{G_0(Z)} \equiv \frac{F_1(Z)}{G_1(Z)} \equiv R(Z) \pmod{Z^n} \\ \implies &F_0(Z)G_1(Z) \equiv F_1(Z)G_0(Z) \pmod{Z^n} \end{aligned}$$

And

$$\begin{aligned} \deg F_0G_1 &= \deg F_0 + \deg G_1 \leq d_F + d_G \leq n \\ \deg F_0G_1 - \deg F_1G_0 &= \deg F_0 + \deg G_1 - \deg F_1 - \deg G_0 \\ &= (\deg F_0 - \deg G_0) + (\deg G_1 - \deg F_1) \\ &= m - m = 0 \end{aligned}$$

Algorithm 3: XYZ-v2 : Decode Function.

We use a queue Q to record all pure cells now;
We use $P[i]$ to record whether i has been verified;
Function `PureCellVerify`(i)**:**
  $m \leftarrow \mathcal{D}^c[i]$;
  if $m > l$ **then**
    $m \leftarrow m - 2l - 1$;
  $(f_0(Z), f_1(Z)) \leftarrow$ `RFuncReconstruct` $(\mathcal{D}^p[i], m)$;
  $\Delta_A \leftarrow (f_0(Z))$;
  $\Delta_B \leftarrow (f_1(Z))$;
  if $|\Delta_A| + |\Delta_B| < \deg f_0(Z) + \deg f_1(Z)$ **then**
    `// Decoding process fails.`
    return False;
  if $\sum_{x\in\Delta_A} h_{fp}(x) - \sum_{x\in\Delta_B} h_{fp}(x) \neq \mathcal{D}^{fp}[i]$ **then**
    `// Fingerprint verification fails`
    return False;
  for $x \in (\Delta_A \cup \Delta_B)$ **do**
    if $\forall j \in [k], h_j(x) \neq i$ **then**
      `// Rehashing verification fails`
      return False;
  return True;
Procedure `Extract`$(x, type)$**:**
  `//` $type = 0$ `for` $x \in A \setminus B$
  `//` $type = 1$ `for` $x \in B \setminus A$.
  for $j \in [k]$ **do**
    if $type = 0$ **then**
      $\mathcal{D}^c[h_j(x)] \leftarrow (\mathcal{D}^c[h_j(x)] - 1) \bmod (2l+1)$;
      $\mathcal{D}^p[h_j(x)] \leftarrow \dfrac{\mathcal{D}^p[h_j(x)]}{Z - x} \bmod Z^l$;
      $\mathcal{D}^{fp}[h_j(x)] \leftarrow \mathcal{D}^{fp}[h_j(x)] - h_{\text{fp}}(x)$;
    else
      $\mathcal{D}^c[h_j(x)] \leftarrow (\mathcal{D}^c[h_j(x)] + 1) \bmod (2l+1)$;
      $\mathcal{D}^p[h_j(x)] \leftarrow \mathcal{D}^p[h_j(x)] \times (Z - x) \bmod Z^l$;
      $\mathcal{D}^{fp}[h_j(x)] \leftarrow \mathcal{D}^{fp}[h_j(x)] + h_{\text{fp}}(x)$;
    if $\neg P[h_j(x)] \wedge$ `PureCellVerify`$(h_j(x))$ **then**
      $P[h_j(x)] \leftarrow$ True;
      Q.`Push`$(h_j(x))$;
Function `Decode`**():**
  $(S_A, S_B) \leftarrow (\varnothing, \varnothing)$;
  `// The answers are recorded in the two sets.`
  for $i \in [M]$ **do**
    if `PureCellVerify` (i) **then**
      Q.`Push`(i);
  while Q *is not empty* **do**
    $i \leftarrow Q$.`Front`$()$;
    $(\Delta_A, \Delta_B) \leftarrow$ `PureCellDecode`(i);
    for $x \in \Delta_A$ **do**
      `Extract`$(x, 0)$
    for $x \in \Delta_B$ **do**
      `Extract`$(x, 1)$
    Q.`Pop`$()$;
    $(S_A, S_B) \leftarrow (S_A \cup \Delta_A, S_B \cup \Delta_B)$;
  if *there exists a cell that has not been decoded* **then**
    `// Decoding fails.`
    return ERROR;
  return (S_A, S_B);

implies that the degrees of polynomials on both sides of the modulo equation are equal and do not exceed n. Since the two polynomials are both monic, it is easy to prove that

$$F_0(Z)G_1(Z) = F_1(Z)G_0(Z)$$

Since F_1, G_1 are coprime, we have

$$G_1(Z) \mid F_0(Z)G_1(Z) \implies G_1(Z) \mid F_1(Z)G_0(Z) \implies G_1(Z) \mid G_0(Z)$$

Similarly, we have $G_0(Z) \mid G_1(Z)$, which yields $G_0(Z) = G_1(Z)$ since they are monic. Symmetrically, there is $F_0(Z) = F_1(Z)$, thus the solution is unique. □

C.2 The Original RFR

Next, we illustrate how to *find a solution* through a series of lemmas, building the argument step by step. We first introduce the original RFR as follows.

Lemma 1. *(The original RFR algorithm,*

Proof. One can see that

$$\frac{F(Z)}{G(Z)} \equiv R(Z) \pmod{Z^N}$$
$$\iff \exists K \in \mathcal{U}[Z], G(Z)R(Z) + K(Z)Z^N = F(Z)$$

Thus, our goal is to find $G(Z)$ and $K(Z)$ such that

$$\deg G \le \lfloor N/2 \rfloor,\ \deg(G(Z)R(Z) + K(Z)Z^N) \le \lfloor N/2 \rfloor$$

Denoting $A := Z^N, B := R(Z)$, we use an approach inspired by the Euclidean algorithm and formulate a recurrence relation as follows:

$$H_0(Z) := A,\ H_1(Z) := B$$
$$\forall i \ge 2, H_i(Z) := H_{i-2}(Z) \bmod H_{i-1}(Z)$$
$$Q_i(Z) := \frac{H_{i-2}(Z) - H_i(Z)}{H_{i-1}(Z)}$$

We continue the definition until $H_i(Z) = 0$. Since $\mathcal{U}[Z]$ is an Euclidean domain, the process must end. It is easy to prove that:

- $\deg H_i > \deg H_{i-1}$
- $Q_i \in \mathcal{U}[Z]; \deg Q_i = \deg H_{i-2} - \deg H_{i-1}$.

We define $G_i := \begin{bmatrix} H_{i-1} \\ H_i \end{bmatrix} \in \mathrm{M}_{2\times 1}(\mathcal{U}[Z])$, obviously there is:

$$G_i = \begin{bmatrix} 0 & 1 \\ 1 & -Q_i(Z) \end{bmatrix} G_{i-1}$$

For convenience we denote $\begin{bmatrix} 0 & 1 \\ 1 & -Q_i(Z) \end{bmatrix}$ by $\varphi(Q_i)$. We find p such that $\deg H_{p-1} > \lfloor N/2 \rfloor, \deg H_p \le \lfloor N/2 \rfloor$, then

$$G_p = \varphi(Q_p)\varphi(Q_{p-1}) \cdots \varphi(Q_2) \cdot \begin{bmatrix} A \\ B \end{bmatrix}$$

We denote the matrix $\varphi(Q_p) \cdots \varphi(Q_2)$ above as $\mathcal{H}_N(A, B)$. One can see that, for every entry $x(Z)$ in the matrix there is

$$\deg x \le \sum_{i=2}^{p} \deg Q_i = \deg H_0 - \deg H_{p-1} \le N - (\lfloor N/2 \rfloor + 1) \le \lfloor N/2 \rfloor$$

If we obtain $\mathcal{H}_N(A,B)$, suppose there is

$$\mathcal{H}_N(A,B) =: \begin{bmatrix} x_0(Z) & y_0(Z) \\ x_1(Z) & y_1(Z) \end{bmatrix}$$
$$\Longrightarrow \begin{bmatrix} H_{p-1}(Z) \\ H_p(Z) \end{bmatrix} = \begin{bmatrix} x_0(Z) & y_0(Z) \\ x_1(Z) & y_1(Z) \end{bmatrix} \begin{bmatrix} Z^N \\ R(Z) \end{bmatrix}$$
$$\Longrightarrow H_p(Z) := x_1(Z)Z^N + y_1(Z)R(Z)$$

Since $\deg H_p, \deg x_1, \deg y_1 \le \lfloor N/2 \rfloor$, the construction is indeed a valid answer.

Next, we illustrate how to efficiently compute $\mathcal{H}_N(A,B)$ for any $A, B \in \mathcal{U}[Z]$, $\deg A, \deg B \le N$, omitting the details of edge cases for brevity. For integer $L > 0$, we divide A, B as

$$\begin{bmatrix} A(Z) \\ B(Z) \end{bmatrix} = Z^L \begin{bmatrix} A_1(Z) \\ B_1(Z) \end{bmatrix} + \begin{bmatrix} A_2(Z) \\ B_2(Z) \end{bmatrix}, s.t. \max(\deg A_2, \deg B_2) < L$$

We suppose $\deg A_1, \deg B_1 \le \lfloor N/2 \rfloor$. Let $M := \mathcal{H}_{\lfloor N/2 \rfloor}(A_1, B_1)$, there is

$$M \begin{bmatrix} A(Z) \\ B(Z) \end{bmatrix} = M \begin{bmatrix} Z^L A_1(Z) + A_2(Z) \\ Z^L B_1(Z) + B_2(Z) \end{bmatrix} = M \begin{bmatrix} A_1(Z) & A_2(Z) \\ B_1(Z) & B_2(Z) \end{bmatrix} \begin{bmatrix} Z^L \\ 1 \end{bmatrix}$$

We denote $M \begin{bmatrix} A_1(Z) & A_2(Z) \\ B_1(Z) & B_2(Z) \end{bmatrix} =: \begin{bmatrix} C_1(Z) & C_2(Z) \\ D_1(Z) & D_2(Z) \end{bmatrix}$, by definition

$$M \begin{bmatrix} A_1(Z) \\ B_1(Z) \end{bmatrix} = \begin{bmatrix} C_1(Z) \\ D_1(Z) \end{bmatrix}$$

Thus the matrix $\begin{bmatrix} C_1(Z) \\ D_1(Z) \end{bmatrix}$ is indeed the G_p in the previous process, leading to

$$\deg C_1 > \lfloor N/4 \rfloor \ge \deg D_1$$

Since

$$\deg C_2, \deg D_2 \le \lfloor N/4 \rfloor + \max(\deg A_2, \deg B_2) < \lfloor N/4 \rfloor + L$$

$$M \begin{bmatrix} A(Z) \\ B(Z) \end{bmatrix} = \begin{bmatrix} C_1(Z) & C_2(Z) \\ D_1(Z) & D_2(Z) \end{bmatrix} \begin{bmatrix} Z^L \\ 1 \end{bmatrix} = \begin{bmatrix} C_1(Z)Z^L + C_2(Z) \\ D_1(Z)Z^L + D_2(Z) \end{bmatrix}$$

We have

$$\deg(C_1(Z)Z^L + C_2(Z)) > \lfloor N/4 \rfloor + L \ge \deg(D_1(Z)Z^L + D_2(Z))$$

The algorithm for computing $\mathcal{H}_N(A,B)$ can be expressed as follows:

(1) Set $L := \lceil N/2 \rceil$ and compute the value of $M_1 = \mathcal{H}_{\lfloor N/2 \rfloor}(A_1, B_1)$ recursively. Then assign $\begin{bmatrix} A(Z) \\ B(Z) \end{bmatrix} \leftarrow M_1 \begin{bmatrix} A(Z) \\ B(Z) \end{bmatrix}$.
The new $B(Z)$ is indeed the $D_1(Z)Z^L + D_2(Z)$ we discussed before, hence $\deg B \le \lfloor N/4 \rfloor + \lfloor N/2 \rfloor \le \lfloor \frac{3}{4}N \rfloor$.
(2) Perform a single step of Euclidean process, *i.e.*

$$\begin{bmatrix} A(Z) \\ B(Z) \end{bmatrix} \leftarrow \begin{bmatrix} B(Z) \\ A(Z) \bmod B(Z) \end{bmatrix}$$

Up to now, $\deg A, \deg B \le \lfloor \frac{3}{4}N \rfloor$ holds.
(3) Set $L := \lceil N/4 \rceil$ and compute the value of $M_2 = \mathcal{H}_{\lfloor N/2 \rfloor}(A, B)$ recursively. One can see it indeed finishes the desired process.
(4) Return $M_2 \cdot \varphi(Q) \cdot M_1$ as result, where Q denotes the quotient in step 2.

The time complexity can be analyzed as

$$T(N) = 2T(N/2) + O(N \log N) \implies T(N) = O(N \log^2 N)$$

□

C.3 Our Modification

Next, we will use Lemma 1 as a tool to gradually construct the desired algorithm.

Lemma 2. *For any polynomial $R(Z)$ that $\deg R < n$ and an integer $m \in [-n+1, 0]$, we can find two polynomials F, G in $O(n \log^2 n)$ time such that*

- $\deg F + \deg G \le n - 1, \deg F - \deg G = m, G$ *is monic.*
- $\dfrac{F(Z)}{G(Z)} \equiv R(Z) \pmod{Z^n}$.

providing that the solution exists.

Proof. Let $d_F := \lfloor \frac{n-1+m}{2} \rfloor, d_G := \lfloor \frac{n-1-m}{2} \rfloor$. By the previous lemma, we find $f, g \in \mathcal{U}[Z]$ such that:

$$\deg f, \deg g \le d_G; \ \frac{f(Z)}{g(Z)} \bmod Z^{2d_G+1} = Z^m R(Z)$$

Note that $Z \nmid \frac{1}{g(Z)}$ must holds, thus

$$Z^m \mid Z^m R(Z) \implies Z^m \mid \frac{f(Z)}{g(Z)} \implies Z^m \mid f(Z)$$

One can see $(\frac{f(Z)}{Z^m}, g(Z))$ is a valid answer. □

To conclude this part, we hereby begin to prove the Theorem 2 in Section 3.

Theorem 0. *(Our Result: modified RFR algorithm) For any polynomial $R(Z)$ with $\deg R < n$, if there are coprime and monic polynomials $F(Z), G(Z)$ with $\deg F + \deg G \le n$ and $\deg F - \deg G = m$, such that:*

$$\frac{F(Z)}{G(Z)} \bmod Z^n = R(Z)$$

Then the possible $F(Z), G(Z)$ are unique, and can be found using $R(Z)$ and m within $O(n \log^2 n)$ time.

Proof. First, we can see that the requirement of coprime is not important in the construction process: if we find a solution (F, G) where they are not coprime but satisfy all other conditions, then $(\frac{F}{\gcd(F,G)}, \frac{G}{\gcd(F,G)})$ constitutes a valid solution.

Through directly factoring out Z from $R(Z)$, we can convert the problem to the case of $Z \nmid R(Z)$. In addition, since changing the numerator and denominator will change the result to $\frac{1}{R(Z)}$, we suppose $m \ge 0$ here.

For the case of $m = 0$, we find $f, g \in \mathcal{U}[Z]$ by the previous lemma such that:

- $\deg f + \deg g < n, \deg f - \deg g = -1, g$ is monic.
- $\dfrac{f(Z)}{g(Z)} \bmod Z^n = R(Z) - 1$.

Then it is easy to prove $(f(Z) + g(Z), g(Z))$ is a valid answer. For the case of $m \ge 1$, we just find f, g such that:

- $\deg f + \deg g < n, \deg f - \deg g = -m + 1, g$ is monic.
- $\dfrac{f(Z)}{g(Z)} \bmod Z^n = \dfrac{1}{R(Z) - Z^m}$.

Then the answer is $(g(Z) + f(Z)Z^m, f(Z))$. □