2024-257 LatticeFold

I. Preperation

.1 notations

$R_q = \mathbb{Z}_q / (X^d + 1)$

For $a = \sum_{i=1}^{d} a_i X^{i-1} \in R_q$, $\vec{a} = (a_1, \cdots, a_m) \in R_q^m$ $A \in R_q^{\kappa \times m}$

vec(·) vectors or matrices (over $\mathbb{Z}_q$)

$$vec(a) = \begin{bmatrix} a_1 \\ \vdots \\ a_d \end{bmatrix} = (a_1, \cdots, a_d)^\top \in \mathbb{Z}_q^d$$ ——— coefficient vector

$vec(\vec{a}) = [vec(a_1), \cdots, vec(a_m)] \in \mathbb{Z}_q^{m \times d}$ a matrix

$fvec(\vec{a})$ concatenation of $vec(\vec{a})$'s row vectors

Rot(·) rotation matrices (over $\mathbb{Z}_q$)

$Rot(a)$

$Rot(A) = [Rot(a_{i,j})]_{\forall i,j}$

$vec(a \cdot b) = Rot(a) \cdot vec(b)$

$fvec(A\vec{f}) = Rot(A)\, fvec(\vec{f})$

RotSum(·) $R_q \times F_{q^\tau}^d \to F_{q^\tau}^d$

for $\beta \in F_{q^\tau}^d$

$RotSum(a, \beta) = Rot(a) \cdot \beta \Rightarrow RotSum(a, vec(b)) = vec(a \cdot b)$

2. Reduction of Knowledge (from R_1 to R_2)

Def. $Setup(1^\lambda) \to pp$

$\langle P(pp, x_1, w_1), V(pp, x_1) \rangle \longrightarrow (x_2, w_2)$

Properties. Completeness:

If $(x_1, w_1) \in R_1$ and V accepts, then $(x_2, w_2) \in R_2$

Knowledge Soundness: Extract w_1

$\exists$ an extractor Ext, given (x_2, w_2),

$Ext(pp, x_1, st) \to w_1$, st $(x_1, w_1) \in R_1$

Public reducibility: Public F 有确定算法可以从 x_1 中提取出 x_2

$\exists$ deterministic PT algorithm F, transcript tr

$F(pp, x_1, tr) = x_2$

$\Pi_1 : R_1 \to R_2$ $\Pi_2 : R_2 \to R_3$ $\Pi_1 \circ \Pi_2 : R_1 \to R_3$

$\Pi_1 : R_1 \to R_3$
$\Pi_2 : R_2 \to R_4$

$\Pi_1 \times \Pi_2 : R_1 \times R_2 \to R_3 \times R_4$

3 Ajtai Commitment

message $|\vec{x}|_\infty < B$

Setup: $A \leftarrow R_q^{\kappa \times m}$

provide binding only

$Com(A, \vec{x})$: $cm = A\vec{x} \bmod q$

II Folding Scheme for Ajtai Com

1. Relations R_{cm} & R_{eval}

$$R_{cm} = \{(pp, cm \in R_q^k ; \vec{f} \in R_q^m): \quad cm = A\vec{f} \wedge |f|_\infty < B\}$$

$$R_{eval} = \left\{ (pp, cm, \vec{r}, v ; \vec{f}) : \begin{array}{l} (pp, cm, \vec{f}) \in R_{cm} \wedge \\ mle[\hat{f}](\vec{r}) = v \end{array} \right\}$$

$NTT(\hat{f}) = vec(\vec{f}) \qquad \hat{f} = (\hat{f}_1, \cdots, \hat{f}_m) \in R_q^m$

then $v = \hat{f}_r$ iff $eq(r, X) = 1$ $\qquad v \in R_q$

像在 F_q 上做 mle 一样

把 F_q 中的元素换成 R_q 中的 $\qquad a = (a_1, \cdots a_m) \in F_q^m$

$$mle[a](X) = \sum_{\forall i} a_i \cdot eq(i, X)$$

$$= \langle a, eq(i, X) \rangle$$

2. Framework

1. Expansion: reduction of knowledge Π_{cm}: $R_{eval}^B \times R_{cm}^B \rightarrow R_{eval}^B \times R_{eval}^B$
2. Decomposition: Π_{dec}: $R_{eval}^B \times R_{eval}^B \rightarrow (R_{eval}^b)^{2k}$
3. Folding: Π_{fold}: $(R_{eval}^b)^{2k} \rightarrow R_{eval}^B$

Warm-up:

prove $|\vec{f}|_\infty < B$ for $\vec{f} \in R_q^m$

$\vec{f} \cdot \left[\circ_{i=1}^{B-1} (\vec{f} - \vec{i}) \cdot (\vec{f} + \vec{i})\right] = \vec{0}$

$\vec{f} \in \mathbb{Z}_q^{md}$

$\vec{i} = (i, i, \cdots, i)$

define $NTT(\hat{f}) = \vec{f}$, $NTT(\hat{i}) = \vec{i}$

$NTT(\hat{f}) \cdot \left[\circ_{i=1}^{B-1} (NTT(\hat{f}) - NTT(\hat{i}))(NTT(\hat{f}) + NTT(\hat{i})\right] = \vec{0}$

go back to R_q

$\hat{f} \cdot \prod_{i=1}^{B-1} (\hat{f} - \hat{i})(\hat{f} + \hat{i}) = \vec{0}$ ✓

Rewrite $R_{cm} = \{(pp, cm \in R_q^k; \vec{f} \in R_q^m): cm = A\vec{f} \wedge \hat{f} \prod_{i=1}^{B-1} (\hat{f} - \hat{i})(\hat{f} - \hat{i}) = 0\}$

Step 1

$R_{eval} = \{(pp, cm, v; \vec{f} \in R_q^m)$ $y = L(\vec{f})$ $cm = A\vec{f} \wedge |\vec{f}|_\infty < B$ $mle[\hat{f}](0) = v\}$

Step 2

$R_{eval} \times R_{eval}$

split: $y = \sum b^i y_i$ $\xrightarrow{y = cm}$ $y_i = L(\vec{f}_i)$ L: homomorphic

$\vec{f} = \sum b^i \vec{f}_i$

$v = \sum b^i v_i$

$\hat{f} = NTT(\vec{f})$ $mle(\hat{f})[r] = v \xrightarrow[\text{NTT运算同构}]{\text{Lemma 3.1}} mle(\hat{f}_i)(r) = v_i$

Input: $(\mathbb{x}; \mathbb{w}) := (\mathsf{cm}; \vec{\mathbf{f}})$.
Output: $(\mathbb{x}_o; \mathbb{w}_o) := ((0^{\log m}, \mathbf{v}, \mathsf{cm}); \vec{\mathbf{f}})$.
The protocol $\langle \mathsf{P}(\mathsf{pp}, \mathbb{x}; \mathbb{w}), \mathsf{V}(\mathsf{pp}, \mathbb{x}) \rangle$**:**

1. $\mathsf{P} \to \mathsf{V}$: P sends V the evaluation $\mathbf{v} := \mathsf{mle}\left[\hat{\mathbf{f}}\right](0^{\log m})$. P生成关系，不做计算
2. V outputs $\mathbb{x}_o := (0^{\log m}, \mathbf{v}, \mathsf{cm})$. P outputs $\mathbb{w}_o := \vec{\mathbf{f}}$.

Figure 1: The protocol Π_{cm} that reduces $\mathcal{R}^B_{\mathsf{cm}}$ to $\mathcal{R}^B_{\mathsf{eval}}$.

Input: $\mathbb{x} := (\vec{\mathbf{r}}, \mathbf{v}, y)$ and $\mathbb{w} := \vec{\mathbf{f}}$.
Output: $[\mathbb{x}_i = (\vec{\mathbf{r}}, \mathbf{v}_i, y_i), \mathbb{w}_i = \vec{\mathbf{f}}_i]_{i=0}^{k-1}$.
The protocol $\langle \mathsf{P}(\mathsf{pp}, \mathbb{x}; \mathbb{w}), \mathsf{V}(\mathsf{pp}, \mathbb{x}) \rangle$**:**

1. $\mathsf{P} \to \mathsf{V}$: Let $\vec{\mathbf{F}} := (\vec{\mathbf{f}}_0, \ldots, \vec{\mathbf{f}}_{k-1}) := \mathsf{split}_{b,k}(\vec{\mathbf{f}})$. P sends V the values $[y_i, \mathbf{v}_i]_{i=0}^{k-1}$ where

$$y_i := \mathcal{L}(\vec{\mathbf{f}}_i), \qquad \mathbf{v}_i := \mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{r}})$$

A·f_i

直接拆开
不改变运算关系

for every $i \in [0, k)$.
2. V checks that $\sum_{i=0}^{k-1} b^i \cdot y_i \stackrel{?}{=} y$, and $\sum_{i=0}^{k-1} b^i \cdot \mathbf{v}_i \stackrel{?}{=} \mathbf{v}$.
3. V outputs $[\mathbb{x}_i = (\vec{\mathbf{r}}, \mathbf{v}_i, y_i)]_{i=0}^{k-1}$. P outputs $[\mathbb{w}_i = \vec{\mathbf{f}}_i]_{i=0}^{k-1}$.

Figure 2: The protocol Π^*_{dec} that reduces $\mathcal{R}^B_{\mathsf{hom}}$ to $(\mathcal{R}^b_{\mathsf{hom}})^k$.

$\mathcal{L}$：同态映射，eg. $\vec{y} = A\vec{f}$　　Π_{dec} 不改变 $\vec{r}$

Figure 2 Π^*_{dec} is a reduction of knowledge

1. Public reducibility　extract x_2

given $x = (\vec{r}, v, y)$ and trans $[y_i, v_i]_{i=0}^{k-1}$, one can output $[\vec{r}, y_i, v_i]_{i=0}^{k-1}$

2. Completeness　$(x_2, w_2) \in R_2$ if V accepts.

(1) V accepts　(as discussed on last page, $\mathcal{L}$ homomorphic, NTT isomorphic)

(2) $(x_2, w_2) \in R_2$ $\begin{cases} \|f_i\|_\infty < b \text{ for all } i \in [0,k) \\ y_i = L(\vec{f}_i) \\ v_i = \text{mle}(\hat{f}_i)[r] \end{cases}$

3. Knowledge Soundness extract w_1 $(\vec{f})$

given $(x_2, w_2) = [(\vec{r}, v_i, y_i); \vec{f}_i]_{i=0}^{k-1}$ output $\vec{f} = \sum b^i \vec{f}_i$

$\|\vec{f}\|_\infty < B$

$b = B^{1/k}$

Step 3 $[x_i = (\vec{r}_i, v_i, y_i)]_{i=1}^{2k}$ $[w_i = \vec{f}_i]_{i=1}^{2k}$ $\|f_i\|_\infty < b$

↳ $x_0 = (\vec{r}_0, v_0, y_0)$ $w_0 = \vec{f}_0$ $\|f_0\|_\infty < B$

$\vec{r}_0 \leftarrow C$, $NTT(v_0) = \sum_{i=1}^{2k} \text{RotSum}(\rho_i, NTT(v_i))$ $y_0 = \sum_{i=1}^{2k} \rho_i y_i$ $w_0 = f_0 = \sum_{i=1}^{2k} \rho_i f_i$

用Suncheck 证明 1. $v_i = \text{mle}[\hat{f}_i](\vec{r}_i)$

2. $\|f_i\|_\infty < b$

define $g_{1,i}(x) = eq(\vec{r}_i, \vec{x})\, \text{mle}[\hat{f}_i](\vec{x})$

define $g_{2,i}(x) = \prod_{j=-(b-1)}^{b-1} (\text{mle}[\hat{f}_i](\vec{x}) - j)$

We have 1. $g_{1,i}(\vec{r}_i) = v_i$

2. $g_{2,i}(x) = 0$ for all x

combine $g_{1,i}$, $g_{2,i}$ into one: $g(x) = \sum_{i=1}^{2k} [\alpha_i g_{1,i}(x) + \mu_i g_{2,i}(x)]$

Thus $\sum_{\forall b} g(b) = \sum_{i=1}^{2k} \alpha_i v_i$ our claim

Parameters: $c \in \mathbb{N}$, $\mathcal{C} := \mathbb{Z}_q \subseteq \mathcal{R}_q$, and a strong sampling set $\mathcal{C}_{\mathsf{small}} \subseteq \mathcal{R}_q$ with expansion factor $\|\mathcal{C}_{\mathsf{small}}\|_{\mathsf{op}} \leq c$ (Defn. 5).

Input: $[\mathbb{x}_i := (\vec{\mathbf{r}}_i, \mathbf{v}_i, y_i)]_{i=1}^{2k}$ and $[\mathbb{w}_i := \vec{\mathbf{f}}_i]_{i=1}^{2k}$.

Output: $\mathbb{x}_o := (\vec{\mathbf{r}}_o, \mathbf{v}_o, y_o)$, $\mathbb{w}_o := \vec{\mathbf{f}}_o$.

The protocol $\langle \mathsf{P}(\mathsf{pp}, \mathbb{x}; \mathbb{w}), \mathsf{V}(\mathsf{pp}, \mathbb{x}) \rangle$**:**

1. $\mathsf{V} \to \mathsf{P}$: V sends P challenges $[\alpha_i]_{i=1}^{2k} \xleftarrow{\$} \mathcal{C}^{2k}$, $[\mu_i]_{i=1}^{2k-1} \xleftarrow{\$} \mathcal{C}^{2k-1}$ and $\vec{\beta} \xleftarrow{\$} \mathcal{C}^{\log m}$.
2. $\mathsf{V} \leftrightarrow \mathsf{P}$: P and V run a sum-check protocol for the claim

$$\sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log m}} g(\vec{\mathbf{b}}) = \sum_{i=1}^{2k} \alpha_i \mathbf{v}_i \,.$$

Set $\mu_{2k} := 1$, here the polynomial $g(\vec{\mathbf{x}}) \in \mathcal{R}_q^{\leq 2b}[X_1, \ldots, X_{\log m}]$ is defined as

$$g(\vec{\mathbf{x}}) := \sum_{i=1}^{2k} \left[\alpha_i g_{1,i}(\vec{\mathbf{x}}) + \mu_i g_{2,i}(\vec{\mathbf{x}})\right] , \tag{14}$$

$$\forall i \in [2k] : g_{1,i}(\vec{\mathbf{x}}) := eq(\vec{\mathbf{r}}_i, \vec{\mathbf{x}}) \cdot \mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{x}}) \,, \tag{15}$$

$$\forall i \in [2k] : g_{2,i}(\vec{\mathbf{x}}) := eq(\vec{\beta}, \vec{\mathbf{x}}) \cdot \prod_{j=-(b-1)}^{b-1} \left(\mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{x}}) - j\right) . \tag{16}$$

The protocol reduces to check the evaluation claim $g(\vec{\mathbf{r}}_o) \stackrel{?}{=} s$ where $\vec{\mathbf{r}}_o \xleftarrow{\$} \mathcal{C}^{\log m}$ is the sum-check challenge vector sampled by V.

3. $\mathsf{P} \to \mathsf{V}$: P sends V values $\left[\theta_i := \mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{r}}_o)\right]_{i=1}^{2k}$.
4. V computes $[\mathbf{e}_i := eq(\vec{\mathbf{r}}_i, \vec{\mathbf{r}}_o)]_{i=1}^{2k}$ and $\mathbf{e}^* := eq(\vec{\beta}, \vec{\mathbf{r}}_o)$ and checks that

$$s \stackrel{?}{=} \sum_{i=1}^{2k} \left[\alpha_i \mathbf{e}_i \theta_i + \mu_i \mathbf{e}^* \cdot \prod_{j=1-b}^{b-1} (\theta_i - j)\right] .$$

5. $\mathsf{V} \to \mathsf{P}$: V sends P random challenge $[\rho_i]_{i=2}^{2k} \xleftarrow{\$} \mathcal{C}_{\mathsf{small}}^{2k-1}$. Set $\rho_1 := 1$.
6. V outputs $\mathbb{x}_o := (\vec{\mathbf{r}}_o, \mathbf{v}_o, y_o)$ where

$$\mathsf{NTT}(\mathbf{v}_o) = \sum_{i=1}^{2k} \mathsf{RotSum}(\rho_i, \mathsf{NTT}(\theta_i)) \,, \qquad y_o := \sum_{i=1}^{2k} \rho_i y_i \,.$$

7. P further outputs $\mathbb{w}_o := \vec{\mathbf{f}}_o = \sum_{i=1}^{2k} \rho_i \cdot \vec{\mathbf{f}}_i$.

Figure 3: The protocol Π_{fold} that reduces $(\mathcal{R}_{\mathsf{bind}}^b)^{2k}$ to $\mathcal{R}_{\mathsf{bind}}^B$.

Figure3 is a reduction of knowledge

1. Public Reducibility

from $[X_i = (\vec{r}_i, v_i, y_i)]_{i=1}^{2k}$ $[w_i = \vec{f}_i]_{i=1}^{2k}$ to $[X_0 = (\vec{r}_0, v_0, y_i)]$ $[w_0 = \vec{f}_0]$

sample challeges $\vec{r}_0$ (sumcheck) set $\vec{r}_0$ $NTT(r_0) = \cdots$

$[\rho_i]_{i=1}^{2k}$ (fold) $y_0 = \sum \rho_i y_i$

2. Completeness

(1) V accepts $s = \sum_{i=1}^{2k} \left[\alpha e_i \theta_i + \mu_i e^* \cdot \prod_{j=1-b}^{b-1} (\theta_i - j) \right]$ g is well-formed

(2) R_2 is satisfied

$(x_0, w_0) = ((\vec{r}_0, v_0, y_0)\ f_0)$ satisfy : $\begin{cases} L(f_0) = y_0 \quad (Af_0 = y_0) \\ mle(f_0)[r_0] = v_0 \\ \|f_0\|_\infty \le B \end{cases}$

Lemma 3.3. *Given any $\ell \in \mathbb{N}$ and a power-of-two $m \in \mathbb{N}$, let $\vec{\mathbf{r}} \in \mathbb{Z}_q^{\log m}$ be a vector and let $\mathcal{L} : \mathcal{R}_q^m \to \mathcal{Y}$ be any $\mathcal{R}_q$-module homomorphism. Given any $[\rho_i]_{i=1}^{\ell} \in \mathcal{R}_q^{\ell}$ and any $[\mathbf{v}_i, y_i; \vec{\mathbf{f}}_i]_{i=1}^{\ell}$ such that $y_i = \mathcal{L}(\vec{\mathbf{f}}_i)$ and $\mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{r}}) = \mathbf{v}_i$ for all $i \in [\ell]$. Set $\mathbf{v}_o, y_o, \vec{\mathbf{f}}_o$ such that*

$$\mathsf{NTT}(\mathbf{v}_o) = \sum_{i=1}^{\ell} \mathsf{RotSum}(\rho_i, \mathsf{NTT}(\mathbf{v}_i)), \qquad y_o := \sum_{i=1}^{\ell} \rho_i \cdot y_i, \qquad \vec{\mathbf{f}}_o := \sum_{i=1}^{\ell} \rho_i \cdot \vec{\mathbf{f}}_i,$$

$\mathbf{v}_o = \sum \mathsf{RotSum}(\rho_i, \mathbf{v}_i)$

where RotSum *is defined in Lemma 2.1. Then we have that $y_o = \mathcal{L}(\vec{\mathbf{f}}_o)$ and $\mathsf{mle}\left[\hat{\mathbf{f}}_o\right](\vec{\mathbf{r}}) = \mathbf{v}_o$.*

线性运算（同态）

proof (1). $L(f_0) = y_0$

(2) $NTT(v_0) = \sum RotSum(\rho_i, NTT(v_i))$

according to Lemma 3.1

since $\text{mle}[\hat{f}_i](\vec{r}) = v_i$, then $\text{mle}[\text{NTT}(\hat{f}_i)](r) = \text{NTT}(v_i)$

单个$\mathcal{R}_q$中的元素在NTT前后有一一对应关系（同构），$\mathcal{R}_q$中的向量在NTT前后也是

Lemma 3.1. *Let $m \in \mathbb{N}$ be a power-of-two and $\mathcal{R}_q \cong \mathbb{F}_{q^\tau}^{d/\tau}$ for some $\tau \in \mathbb{N}$ where $\tau \mid d$. For any $\vec{\mathbf{f}} \in \mathcal{R}_q^m$ and any vector $\vec{\mathbf{r}} \in \mathcal{R}_q^{\log m}$ such that $\text{NTT}(\vec{\mathbf{r}}) = \underbrace{(\vec{\mathbf{r}}^*, \ldots, \vec{\mathbf{r}}^*)}_{d/\tau}$ (where $\vec{\mathbf{r}}^* \in \mathbb{F}_{q^\tau}^{\log m}$),*

$\vec{r} \in \{0,1\}^{\log m}$ 常数的NTT是重复的常数

let $\hat{\mathbf{f}} := (\hat{\mathbf{f}}_1, \ldots, \hat{\mathbf{f}}_\tau) \in \mathcal{R}_q^{m \times \tau}$ denote the vector such that $\text{NTT}(\hat{\mathbf{f}}) = \text{vec}(\vec{\mathbf{f}})$. We have that $\text{mle}\left[\hat{\mathbf{f}}\right](\vec{\mathbf{r}}) \cong \text{mle}\left[\text{vec}(\vec{\mathbf{f}})\right](\vec{\mathbf{r}}^*)$ where

$$\text{mle}\left[\hat{\mathbf{f}}\right](\vec{\mathbf{r}}) := \left(\text{mle}\left[\hat{\mathbf{f}}_1\right](\vec{\mathbf{r}}), \ldots, \text{mle}\left[\hat{\mathbf{f}}_\tau\right](\vec{\mathbf{r}})\right) \in \mathcal{R}_q^\tau,$$

NTT前的Eval　同构

$$\text{mle}\left[\text{vec}(\vec{\mathbf{f}})\right](\vec{\mathbf{r}}^*) := \left(\text{mle}\left[\text{vec}_1(\vec{\mathbf{f}})\right](\vec{\mathbf{r}}^*), \ldots, \text{mle}\left[\text{vec}_d(\vec{\mathbf{f}})\right](\vec{\mathbf{r}}^*)\right) \in \mathbb{F}_{q^\tau}^d.$$

NTT后的Eval

Recall that $\text{mle}[\cdot]$ denotes multilinear extensions (Defn. 2.4) and $\text{vec}_i(\vec{\mathbf{f}}) \in \mathbb{Z}_q^m$ is the i-th $(1 \le i \le d)$ column of the coefficient embedding matrix $\text{vec}(\vec{\mathbf{f}})$.

Proof. By definition of $\hat{\mathbf{f}}$, we have that

$$\text{NTT}(\hat{\mathbf{f}}) = (\text{NTT}(\hat{\mathbf{f}}_1), \ldots, \text{NTT}(\hat{\mathbf{f}}_\tau)) = (\text{vec}_1(\vec{\mathbf{f}}), \ldots, \text{vec}_d(\vec{\mathbf{f}})). \tag{9}$$

Also observe that

$$\left(\text{mle}\left[\hat{\mathbf{f}}_1\right](\vec{\mathbf{r}}), \ldots, \text{mle}\left[\hat{\mathbf{f}}_\tau\right](\vec{\mathbf{r}})\right) = \left(\left\langle \hat{\mathbf{f}}_j, \bigotimes_{i=1}^{\log m} (\vec{\mathbf{r}}_i, 1 - \vec{\mathbf{r}}_i) \right\rangle\right)_{j=1}^{\tau} \tag{10}$$

tensor(r)　mle operation

where $\bigotimes$ denotes tensor product over $\mathcal{R}_q$. Thus the lemma holds by the Chinese Remainder Theorem. □

Thus $\text{NTT}(v_0) = \sum \text{RotSum}(\rho_i, \text{NTT}(v_i)) = \sum \text{RotSum}(\rho_i, \text{mle}[\text{NTT}(\hat{f}_i)](\vec{r}))$

$= \sum \text{RotSum}(\rho_i, \langle \text{NTT}(f_i), \otimes r_i(1-r_i) \rangle)$

$= \sum \langle \text{RotSum}(\rho_i \cdot \text{NTT}(f_i)), \otimes r_i(1-r_i) \rangle = \sum \langle \text{vec}(\rho_i \cdot \text{NTT}(f_i)), \otimes r_i(1-r_i) \rangle$

$= \langle \text{vec}(\sum \rho_i \text{NTT}(f_i)), \text{tensor}(\vec{r}) \rangle = \langle \text{NTT}(f_0), \text{tensor}(\vec{r}) \rangle$

$v_0 = \text{NTT}^{-1} \langle \text{NTT}(f_0), \text{tensor}(\vec{r}) \rangle$

直接用 v_0 和 $\sum_{i=1}^{l} \text{RotSum}(\rho_i, v_i)$ 写也可以得到类似的结果，但需要转换vectors与polynomials

(3) $\|f_0\|_\infty \le B = b^k$

$$\|f_0\|_\infty = \|\sum_{i=1}^{2k} \rho_i \vec{f}_i\|_\infty \le \sum \|\rho_i f_i\|_\infty \le \sum c\,\|f_i\|_\infty \le \sum_{i=1}^{2k} c\cdot(b-1) = 2kc(b-1)$$

by choosing c properly, it holds.

3. Knowledge Soundness

Ext 作为 V 与 A、P 交互以提取 $\vec{f}_i$

A 产生初始 input, P 执行协议产生 proof

用 P_1，当 P 产生一组满足条件的 f_0 时，使用 randomness ρ_2 rewind P

因此，可以得到 2 组 $f_0^{(1)}$、$f_0^{(2)}$ 满足 $\begin{cases} f_1 + \rho_1 f_2 = f_0^{(1)} \\ f_1 + \rho_2 f_2 = f_0^{(2)} \end{cases}$ 则可求解 f_1、f_2

requirement: ρ_1、ρ_2 are small and invertible (lattice 常用的 challenge space)

Ⅲ Latticefold for CCS

1. CCS over Rings

1. CCS constructure

R1CS $Az \circ Bz = Cz$

$-1Cz + (Az \circ Bz) = \vec{0}$

- size bounds $m, n, l, t, q, d \in \mathbb{N}$, $n > l$
- matrices $\{M_j \in F^{m\times n}\}_{j\in[t]}$
- q 个 multisets $\{S_i\}_{i\in[q]}$

  S_i 中每个元素都是 $1, \cdots, t$ 组成，至多 d 个，例如 $(1, 2, 3, \cdots, d)$，若 $d < t$
- q 个 constants $\{c_i\}_{i\in[q]}$, $c_i \in F$

一个 CCS instance (S, x) 满足 $\sum_{i\in[q]} c_i \circ_{j\in[S_i]} (M_j \cdot \vec{z}) = \vec{0}$，其中 $z = (w, 1, x) \in F^n$

$w \in F^{n-l-1}$, $x \in F^l$

2. CCS over Rings

$$\sum_{i=1}^{n_s} c_i \circ_{j\in[s_i]} (M_j \cdot \vec{z}) = \vec{0} \qquad pp = (n_r, n_c, t, n_s, deg, l_{in})$$

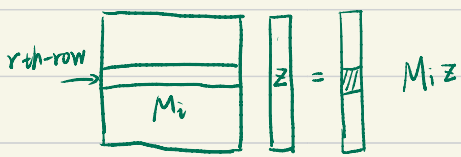
其中 $c_i \in R$, $M_j \in R^{n_r \times n_c}$, $z = (x, 1, w) \in R^{n_c}$

Linear CCS (mle) $\quad x_{lccs} = (\vec{r}, [u_i]_{i=1}^{t}, x_{ccs}, h), \quad w = w_{ccs}$

$$u_i = \sum_{b\in\{0,1\}^{\log n_c}} mle[M_i](\vec{r}, \vec{b}) \cdot mle[z](\vec{b})$$

mle[z] 取遍
mle[Mi] 只取 r 总

其中 $z = (x, h, w)$



2. committed CCS

把 CCS 中的 z 用 gadget matrix 作 B 进制分解，使得每个分量都小于 B

用 Ajtai 承诺对分解后的 $\vec{z}$ (即 $\vec{f}$) 承诺

分别说明 { z 满足 CCS 约束、f 满足承诺等式
{ z 是正确分解的

$$R_{cmccs} = \left\{ \left(\begin{array}{l} pp, x = (cm \in R_q^k, x_{ccs} \in R_q^{t \cdot in}); \\ w = (\vec{f} \in R_q^m, w_{ccs} \in R_q^{n_c - l_{in} - 1}) \end{array} \right) \middle| \begin{array}{l} (pp_{cm}, cm; f) \in R_{cm}^B \\ \wedge (pp_{ccs}, x_{ccs}; w_{ccs}) \in R_{ccs} \\ \wedge z_{ccs} = G\vec{f} \end{array} \right\}$$

Evaluation CCS

分别表示 f 的 mle 计算和 CCS 约束中 z 的 mle 计算，用 gadget matrix 连接起来

$$R^{B}_{evalccs} = \left\{ \begin{array}{ll} pp, x=(\vec{r}, cm, v, [u_i]_{i=1}^{t}, x_{ccs}, h); & z_{ccs}=Gf \wedge (pp_{cm}, (cm, v, \vec{r}), \vec{f}) \in R^{B}_{eval} \\ w=(\vec{f}, w_{ccs}) & \wedge (pp_{ccs}, (\vec{r}_{ccs}, [u_i]_{i=1}^{t}, x_{ccs}, h), w_{ccs}) \in R_{lccs} \end{array} \right\}$$

3. Folding

Step1 : from commited CCS to eval CCS

Step2 : Decomposition

Step3 : Folding

Step 1 $R_{evalccs} \times R_{cmccs} \longrightarrow R_{evalccs} \times R_{evalccs}$

Apply sumcheck

1、定义 g(x) 为 CCS 约束的多项式，则 g(x)=0，不论 x 是任何取值

$$g(\vec{x}) := eq(\vec{\beta}, \vec{x}) \cdot \left(\sum_{i=1}^{n_s} c_i \cdot \left[\prod_{j \in S_i} \left(\sum_{\vec{b} \in \{0,1\}^{\log n_c}} \text{mle}[M_j](\vec{x}, \vec{b}) \cdot \text{mle}[z_{ccs}](\vec{b}) \right) \right] \right).$$

求 c_i 的和　　(表示 $M_j z$ 在 x 点取值)

理解 g(x)　　取一个 ccs instance　　$c_1 M_1 z + c_2 (M_2 z \circ M_3 z) = \vec{0}$

表示 ccs 约束计算的向量在 x 处的 entry

$\Rightarrow$ g(x)=0

2、做 evaluation 的转化

3、证明 2 的转化是正确的（通过 Sumcheck 的等式说明）

Parameters: A strong sampling set $\mathcal{C} := \mathbb{Z}_q \subseteq \mathcal{R}_q$ (Defn. 2.3).
Input: $\mathbb{x} := (\mathsf{cm}, \mathbb{x}_{\mathsf{ccs}}) \in \mathcal{R}_q^{\kappa} \times \mathcal{R}_q^{\ell_{\mathsf{in}}}$ and $\mathbb{w} := (\vec{\mathbf{f}}, \mathbb{w}_{\mathsf{ccs}}) \in \mathcal{R}_q^{m} \times \mathcal{R}_q^{n_c - \ell_{\mathsf{in}} - 1}$.
Output: $\mathbb{x}_o := (\vec{\mathbf{r}}_o \in \mathcal{R}_q^{\log m}, \mathbf{v} \in \mathcal{R}_q, \mathsf{cm}, [\mathbf{u}_i \in \mathcal{R}_q]_{i=1}^{t}, \mathbb{x}_{\mathsf{ccs}}, 1)$ and $\mathbb{w}_o := (\vec{\mathbf{f}}, \mathbb{w}_{\mathsf{ccs}})$.
The protocol $\langle \mathsf{P}(\mathsf{pp}, \mathbb{x}; \mathbb{w}), \mathsf{V}(\mathsf{pp}, \mathbb{x}) \rangle$**:**

1. $\mathsf{V} \to \mathsf{P}$: V sends P a random vector $\vec{\beta} \xleftarrow{\$} \mathcal{C}^{\log m}$.
2. $\mathsf{P} \leftrightarrow \mathsf{V}$: P and V run a sum-check protocol for the claim $\sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log m}} g(\vec{\mathbf{b}}) = 0$. Let $\mathbf{z}_{\mathsf{ccs}} := (\mathbb{x}_{\mathsf{ccs}}, 1, \mathbb{w}_{\mathsf{ccs}})$. $g \in \mathcal{R}_q^{\leq \mathsf{deg}+1}[X_1, \ldots, X_{\log m}]$ is defined as[a]

$$g(\vec{\mathbf{x}}) := eq(\vec{\beta}, \vec{\mathbf{x}}) \cdot \left(\sum_{i=1}^{n_s} c_i \cdot \left[\prod_{j \in S_i} \left(\sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log n_c}} \mathsf{mle}\left[M_j\right](\vec{\mathbf{x}}, \vec{\mathbf{b}}) \cdot \mathsf{mle}\left[\mathbf{z}_{\mathsf{ccs}}\right](\vec{\mathbf{b}}) \right) \right] \right) .$$

   The protocol reduces to a random evaluation check $g(\vec{\mathbf{r}}_o) \stackrel{?}{=} s$ for some $s \in \mathcal{R}_q$, and $\vec{\mathbf{r}}_o \xleftarrow{\$} \mathcal{C}^{\log m}$ is the sum-check challenge vector sampled by V.
3. $\mathsf{P} \to \mathsf{V}$: P sends V the values $(\mathbf{v}, [\mathbf{u}_i]_{i=1}^{t})$ where $\mathbf{v} := \mathsf{mle}\left[\hat{\mathbf{f}}\right](\vec{\mathbf{r}}_o)$ and for every $i \in [t]$, $\mathbf{u}_i$ is computed as

$$\mathbf{u}_i := \sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log n_c}} \mathsf{mle}\left[M_i\right](\vec{\mathbf{r}}_o, \vec{\mathbf{b}}) \cdot \mathsf{mle}\left[\mathbf{z}_{\mathsf{ccs}}\right](\vec{\mathbf{b}}) .$$

4. V computes $\mathbf{e} := eq(\vec{\beta}, \vec{\mathbf{r}}_o)$ and checks that

$$\mathbf{e} \cdot \left(\sum_{i=1}^{n_s} c_i \cdot \prod_{j \in S_i} \mathbf{u}_j \right) \stackrel{?}{=} s .$$

5. V outputs $\mathbb{x}_o := (\vec{\mathbf{r}}_o, \mathbf{v}, \mathsf{cm}, [\mathbf{u}_i]_{i=1}^{t}, \mathbb{x}_{\mathsf{ccs}}, 1)$. P outputs $\mathbb{w}_o := (\vec{\mathbf{f}}, \mathbb{w}_{\mathsf{ccs}})$.

[a] deg is the CCS gate degree.

Figure 4: The protocol Π_{ccs} that reduces $\mathcal{R}_{\mathsf{cmccs}}^{B}$ to $\mathcal{R}_{\mathsf{evalccs}}^{B}$.

Step 2 在 mle[f](r) 拆分后，加上了 CCS 的计算

用拆分的 $\vec{f}_i$ 做后续 CCS 计算（由于 z 与 f 是一致的）

Input: $\mathbb{x} := (\vec{\mathbf{r}}, \mathbf{v}, y, \mathbf{u}, \mathbb{x}_w)$ and $\mathbb{w} := (\vec{\mathbf{f}}, \vec{\mathbf{w}})$.
Output: $[\mathbb{x}_i = (\vec{\mathbf{r}}, \mathbf{v}_i, y_i, \mathbf{u}_i, \mathbb{x}_{w,i}), \mathbb{w}_i = (\vec{\mathbf{f}}_i, \vec{\mathbf{w}}_i)]_{i=0}^{k-1}$.
The protocol $\langle \mathsf{P}(\mathsf{pp}, \mathbb{x}; \mathbb{w}), \mathsf{V}(\mathsf{pp}, \mathbb{x}) \rangle$**:**

1. P computes $(\vec{\mathbf{f}}_0, \ldots, \vec{\mathbf{f}}_{k-1}) \leftarrow \mathsf{split}_{b,k}(\vec{\mathbf{f}})$ (Eqn. 12).
2. $\mathsf{P} \to \mathsf{V}$: P sends V the values $[y_i, \mathbf{v}_i, \mathbf{u}_i, \mathbb{x}_{w,i}]_{i=0}^{k-1}$:

$$y_i := \mathcal{L}(\vec{\mathbf{f}}_i)\,, \qquad \mathbf{v}_i := \mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{r}})\,,$$

$$\mathbf{u}_i := \left\langle M \cdot \mathsf{tensor}(\vec{\mathbf{r}}), \mathcal{L}_w(\vec{\mathbf{f}}_i) \right\rangle\,, \qquad \mathbb{x}_{w,i} := \mathcal{L}_w(\vec{\mathbf{f}}_i)[1, n_{\mathsf{in}}]$$

   for every $i \in [0, k)$.
3. V checks that $\sum_{i=0}^{k-1} b^i \cdot [y_i, \mathbf{v}_i, \mathbf{u}_i, \mathbb{x}_{w,i}] \stackrel{?}{=} [y, \mathbf{v}, \mathbf{u}, \mathbb{x}_w]$.
4. V outputs $[\mathbb{x}_i = (\vec{\mathbf{r}}, \mathbf{v}_i, y_i, \mathbf{u}_i, \mathbb{x}_{w,i})]_{i=0}^{k-1}$. P outputs $[\mathbb{w}_i = (\vec{\mathbf{f}}_i, \mathcal{L}_w(\vec{\mathbf{f}}_i))]_{i=0}^{k-1}$.

Figure 5: The protocol Π^*_{ccsdec} that reduces $\mathcal{R}^B_{\mathsf{ccshom}}$ to $(\mathcal{R}^b_{\mathsf{ccshom}})^k$.

Parameters: $c \in \mathbb{N}$, $\mathcal{C} := \mathbb{Z}_q \subseteq \mathcal{R}_q$ and a strong sampling set $\mathcal{C}_{\mathsf{small}}$ with expansion factor $\|\mathcal{C}_{\mathsf{small}}\|_{\mathsf{op}} \leq c$.
Input: $\mathbb{x} := [\mathbb{x}_i := (\vec{\mathbf{r}}_i, \mathbf{v}_i, y_i, \mathbf{u}_i, \mathbb{x}_{w,i})]_{i=1}^{2k}$ and $\mathbb{w} := [\mathbb{w}_i := (\vec{\mathbf{f}}_i, \vec{\mathbf{w}}_i)]_{i=1}^{2k}$.
Output: $\mathbb{x}_o := (\vec{\mathbf{r}}_o, \mathbf{v}_o, y_o, \mathbf{u}_o, \mathbb{x}_{w,o})$, $\mathbb{w}_o := (\vec{\mathbf{f}}_o, \vec{\mathbf{w}}_o)$.
The protocol $\langle \mathsf{P}(\mathsf{pp}, \mathbb{x}; \mathbb{w}), \mathsf{V}(\mathsf{pp}, \mathbb{x}) \rangle$**:**

1. $\mathsf{V} \to \mathsf{P}$: V sends P $[\alpha_i, \zeta_i]_{i=1}^{2k} \xleftarrow{\$} (\mathcal{C} \times \mathcal{C})^{2k}$, $[\mu_i]_{i=1}^{2k-1} \xleftarrow{\$} \mathcal{C}^{2k-1}$, and $\vec{\beta} \xleftarrow{\$} \mathcal{C}^{\log m}$.
2. $\mathsf{V} \leftrightarrow \mathsf{P}$: P and V run a sum-check protocol for the claim

$$\sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log m}} g(\vec{\mathbf{b}}) = \sum_{i=1}^{2k} (\alpha_i \mathbf{v}_i + \zeta_i \mathbf{u}_i) \,.$$

Set $\mu_{2k} := 1$, here the polynomial $g(\vec{\mathbf{x}}) \in \mathcal{R}_q^{\leq 2b}[X_1, \ldots, X_{\log m}]$ is defined as

$$g(\vec{\mathbf{x}}) := \sum_{i=1}^{2k} [\alpha_i g_{1,i}(\vec{\mathbf{x}}) + \mu_i g_{2,i}(\vec{\mathbf{x}}) + \zeta_i g_{3,i}(\vec{\mathbf{x}})] \,, \tag{30}$$

where for all $i \in [2k]$,

$$g_{1,i}(\vec{\mathbf{x}}) := eq(\vec{\mathbf{r}}_i, \vec{\mathbf{x}}) \cdot \mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{x}}) \,, \quad g_{2,i}(\vec{\mathbf{x}}) := eq(\vec{\beta}, \vec{\mathbf{x}}) \cdot \prod_{j=-(b-1)}^{b-1} \left(\mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{x}}) - j\right) \,,$$

$$g_{3,i}(\vec{\mathbf{x}}) := eq(\vec{\mathbf{r}}_i, \vec{\mathbf{x}}) \cdot \left(\sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log(n+n_{\mathsf{in}})}} \mathsf{mle}[M_i](\vec{\mathbf{x}}, \vec{\mathbf{b}}) \cdot \mathsf{mle}[\mathbf{z}_i](\vec{\mathbf{b}}) \right) .$$

Here $\mathbf{z}_i := (\mathbb{x}_{w,i} || \vec{\mathbf{w}}_i)$ for all $i \in [2k]$.The protocol reduces to check the evaluation claim $g(\vec{\mathbf{r}}_o) \stackrel{?}{=} s$ where $\vec{\mathbf{r}}_o \xleftarrow{\$} \mathcal{C}^{\log m}$ is the sum-check challenge sampled by V.

3. $\mathsf{P} \to \mathsf{V}$: P sends V values $\left[\theta_i := \mathsf{mle}\left[\hat{\mathbf{f}}_i\right](\vec{\mathbf{r}}_o), \eta_i\right]_{i=1}^{2k}$, where for all $i \in [2k]$,

$$\eta_i := \sum_{\vec{\mathbf{b}} \in \{0,1\}^{\log(n+n_{\mathsf{in}})}} \mathsf{mle}[M_i](\vec{\mathbf{r}}_o, \vec{\mathbf{b}}) \cdot \mathsf{mle}[\mathbf{z}_i](\vec{\mathbf{b}}) \,.$$

4. V computes $[\mathbf{e}_i := eq(\vec{\mathbf{r}}_i, \vec{\mathbf{r}}_o)]_{i=1}^{2k}$ and $\mathbf{e}^* := eq(\vec{\beta}, \vec{\mathbf{r}}_o)$ and checks that

$$s \stackrel{?}{=} \sum_{i=1}^{2k} \left[\alpha_i \mathbf{e}_i \theta_i + \mu_i \mathbf{e}^* \cdot \prod_{j=1-b}^{b-1} (\theta_i - j) + \zeta_i \mathbf{e}_i \eta_i \right] .$$

5. $\mathsf{V} \to \mathsf{P}$: V sends P random challenges $[\rho_i]_{i=2}^{2k} \xleftarrow{\$} \mathcal{C}_{\mathsf{small}}^{2k-1}$. Set $\rho_1 := 1$.
6. V output $\mathbb{x}_o := (\vec{\mathbf{r}}_o, \mathbf{v}_o, y_o, \mathbf{u}_o, \mathbb{x}_{w,o})$ where $\mathsf{NTT}(\mathbf{v}_o) = \sum_{i=1}^{2k} \mathsf{RotSum}(\rho_i, \mathsf{NTT}(\theta_i))$ and $[y_o, \mathbf{u}_o, \mathbb{x}_{w,o}] := \sum_{i=1}^{2k} \rho_i \cdot [y_i, \eta_i, \mathbb{x}_{w,i}]$.
7. P outputs $\vec{\mathbf{f}}_o = \sum_{i=1}^{2k} \rho_i \cdot \vec{\mathbf{f}}_i$ and $\vec{\mathbf{w}}_o := \mathcal{L}_w(\vec{\mathbf{f}}_o)[n_{\mathsf{in}} + 1, n_{\mathsf{in}} + n]$.

Figure 6: The protocol Π_{ccsfold} that reduces $(\mathcal{R}_{\mathsf{ccsbind}}^b)^{2k}$ to $\mathcal{R}_{\mathsf{ccsbind}}^B$. We set the $\mathcal{R}_q$-module $\mathcal{U}$ to be $\mathcal{U} := \mathcal{R}_q$. The protocol naturally extends to any $\mathcal{U} := \mathcal{R}_q^t$ where $t > 1$.