密码学 zk 系列

第 10 课：zkSnark-UltraPlonk 证明系统

lynndell 博士

新火科技 密码学专家 lynndell2010@gmail.com

目录

Plonk 证明系统 = 电路 + 多项式承诺

电路：将任意算法转换为多项式。
多项式承诺：发送方证明多项式是正确的，验证方检测多项式是否正确。
Plonk = 标准 plonk 门 + 多项式承诺优化
UltraPlonk = 电路优化（查找表与定制门）+ 多项式承诺

保密数据表 Private Data			PubData 公开数据表 1：异或运算			PubData 公开数据表 2：与运算		
ai	bi	ci	T0	T1	T2	T3	T4	T5
a0	b0	c0	0	0	0	0	0	0
a1	b1	c1	1	0	1	1	0	0
a2	b2	c2	0	1	1	0	1	0
a3	b3	c3	1	1	0	1	1	1
a4	b4	c4						
a5	b5	c5						
a6	b6	c6						

（1）有大量的数据是保密数据，如{ai,bi,ci},i=0,1,2,3,4,5
（2）公开数据表（异或运算、与运算）中的数据与运算规则是公开且正确。

步骤 1：查找表

使用累加器证明：

- 第0行保密数据{a0,b0,c0}属于公开数据表1，则{a0,b0,c0}满足异或运算，而再不需要对该行数据写4个R1CS约束（或8个标准plonk门+5个线约束）

$$(1-a)\times a=0$$
$$(1-b)\times b=0$$
$$(1-c)\times c=0$$
$$(2a)\times(b)=(a+b-c)$$

- 第1行保密数据{a1,b1,c1}属于公开数据表2，则{a1,b1,c1}满足与运算，而不再需要对该行数据写电路约束；

因此，查找表能够节约大量的电路约束。

步骤 2：定制门

需要一个额外的选择多项式，选出数据{a1,b2,c3}，满足乘法约束 a1*b2=c3，则实现了对该组数据的约束。

步骤 3：标准门：

如果没有定制门，仅有**标准门**，则需要使用一个新的标准门进行约束 a6*b6=c6。
还要线约束（也称为置换约束），确保 a6=a1, b6=b2, c6=c3。
因此，使用 1 个标准门和 3 个线约束，实现{a1,b2,c3}的乘法约束。
因此，保密数据表行数增加 1 行、线约束增加 3 个。
因此，定制门能够节约大量的门约束与线约束，降低重复性。

1 Plookup 查找表原理

问题： 以太坊使用哈希函数 SHA256/SHA3 计算哈希值，为降低 gas 费，zkRollup 也应使用 SHA256/SHA3 计算数据摘要。但是，SHA256/SHA3 涉及大量的布尔运算和循环操作，需要把这些**布尔运算**和**循环操作**转换为等价的大素数域上的 Plonk 门。因此，需要 2.5 万个的 Plonk 门才能够实现等价功能的 SHA256/SHA3，这会导致证明生成缓慢。

解决方案：对于使用频率较高的布尔运算和循环操作，
（1）预计算一个公开且正确的输入与输出表格 Table。
（2）证明方$\mathcal{P}$使用**累加器**证明**保密数据**在该表格 Table 中，则等价于证明：输入和输出的运算是正确的。

因此，利用查找表中值的**相等性检测（累加器）**，代替大量的门约束，降低到 0.5 万个 Plonk 门，能够提高证明方$\mathcal{P}$的证明速度。

1.1 预备知识

1.1.1 Schwartz–Zippel 引理

令P为有限域$\mathbb{F}$上的 n 元多项式$P = F(x_1, \ldots, x_n)$，其阶为d。令S为有限域$\mathbb{F}$的子集，从S中选择随机数$r_1, \ldots, r_n$，则多项式等于零的概率可忽略，即

$$\Pr[P(r_1, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

在单变量情况下，一元多项式的阶为d，则最多有d个根。

1.1.2 乘积归约定理

乘积一致性检测原理（**累加器/累乘器**）**：** 给定有限域$\mathbb{F}$上的多项式f的系数$a_1, \ldots, a_n$与多项式g的系数$b_1, \ldots, b_n$和一个子集$H = \{x_1, \ldots, x_n\} \subset \mathbb{F}$。

（1）：如果两个集合$\{a_i\}, i = 1, \ldots, n$和$\{b_i\}, i = 1, \ldots, n$中的元素对应相等$a_i = b_i, i \in [n]$，则乘积值一定相等$\prod_{i \in [n]} a_i = \prod_{i \in [n]} b_i$。

（2）：反之，如果累加器/累乘器的相等$\prod_{i \in [n]} a_i = \prod_{i \in [n]} b_i$，则根据 Schwartz–Zippel 引理，能够构造多项式

$$P(X_1, \ldots, X_n) := \prod_{i \in [n]} (X_i - a_i) = 0$$

在子集H上**随机选择**$(b_1, \ldots, b_n)$使得多项式$P(X_1, \ldots, X_n)$**等于零**的概率可忽略。
猜对一个或 n 个概率都是零。
因此，如果多项式$P(X_1, \ldots, X_n)$等于零，则$(b_1, \ldots, b_n)$是解，所以两个集合中的元素对应相等$a_i = b_i, i \in [n]$。因此，能够推导出多项式f与多项式g相等。

1.1.3 多集合相等性检测

通过添加一个**随机数**，则能够把乘积归约定理转换为一个更有用的工具：**多集合相等性检测**。给定两个向量$\vec{a} = (a_1, \ldots, a_n), \vec{b} = (b_1, \ldots, b_n)$，检测两个向量是否包含相同的元素，甚至以不同的顺序计算重复性。

多集合相等性到乘积一致性的归约：验证方$\mathcal{V}$选择随机数$\gamma \in \mathbb{F}$，并对**随机转移向量**$a' \triangleq a+\gamma, b' \triangleq b+\gamma$（$\gamma$被添加到所有的坐标中），检测**累加器/累乘器**

$$\prod_{i\in[n]} (a_i+\gamma) = \prod_{i\in[n]} (b_i+\gamma)$$

对于多项式中的γ，由 Schwarz-Zippel 引理表明，除非a'和b'两个集合是相等的，否则上述等式成立的概率可忽略。

1.1.4 置换与多集合相等性检测（线约束）

置换：两个值的**位置可交换**而不影响累加器的计算结果。
因此，如果**累加器**的值相等，则迫使两个值相等。置换映射$\sigma: [n] \to [n]$。

关键结论：两个向量集合$\{(a_i, i)_{i\in[n]}\}, \{(b_i, \sigma(i))_{i\in[n]}\}$相等当且仅当$a' = \sigma(b')$。

证明：
（1）选择随机数β，定义向量a', b'

$$a'_i \triangleq a_i + \beta \cdot i,$$
$$b'_i \triangleq b_i + \beta \cdot \sigma(i)$$

如果上述向量集合的乘积值不相等

$$\prod_{i\in[n]} a'_i \neq \prod_{i\in[n]} b'_i$$

则$\{a'\}$和$\{b'\}$元素集合相等的概率可忽略。

（2）反之，如果$\prod_{i\in[n]} a'_i = \prod_{i\in[n]} b'_i$，则根据 Schwarz-Zippel 引理，$\{a'\}$和$\{b'\}$元素集合对应不等的概率可忽略。

1.2 Plookup 核心技术

1.2.2 Plookup 多项式表达

关键结论：

- **多项式协议在代数群上使用多项式承诺、随机打开点、商多项式承诺代替可信第三方$\mathcal{I}$，则三方交互协议变为两方交互协议。**
- **使用 Fait-Shamir 变换计算哈希值，将两方交互协议转换为非交互协议。**

1.2.2.1 多项式协议

证明方$\mathcal{P}$发送一个d阶多项式给可信第三方$\mathcal{I}$（$\mathcal{P}$多项式承诺）。当协议运行完毕，验证方$\mathcal{V}$询问可信第三方$\mathcal{I}$：定义在集合H上的**d阶保密输入多项式**和**公开且正确的预计算多项式**是否满足某种性质（**等价于：**$\mathcal{V}$检查**商**多项式是否存在，等价于检测随机打开点是否正确）。对于固定整数d, D, t, l和$H \subset \mathbb{F}$。

1. 公开且正确的预计算$t_1, \ldots, t_l \in \mathbb{F}_d[X]$和初始化多项式$g_1, \ldots, g_l \in \mathbb{F}_d[X]$。

2. 证明方$\mathcal{P}$与可信第三方$\mathcal{I}$之间的消息是**保密**多项式形式$f\in\mathbb{F}_d[X]$。（$\mathcal{P}$多项式承诺）
3. 验证方$\mathcal{V}$发送随机数α给证明方$\mathcal{P}$。（$\mathcal{P}$计算随机数）
4. 协议执行完毕，证明方$\mathcal{P}$将多项式$f_1,\dots,f_t$发送给可信第三方$\mathcal{I}$。（$\mathcal{P}$多项式随机打开值与商多项式承诺）

验证方$\mathcal{V}$询问可信第三方$\mathcal{I}$，多项式$f_1,\dots,f_t,g_1,\dots,g_l$在$H$范围内对随机数$\alpha$是否满足某个运算关系

$$F(X)=G(\alpha,X,h_1(v_1(X)),\dots,h_M(v_M(X)))\equiv 0$$

$h_i\in\{f_1,\dots,f_t,g_1,\dots,g_l\}$，$G\in\mathbb{F}_d[X,X_1,\dots,X_M]$，$v_1,\dots,v_M\in\mathbb{F}_d[X]$，使得$F\in\mathbb{F}_{<D}[X]$。（等价于：$\mathcal{V}$验证商多项式是否存在，等价于验证多项式的值是否正确）
当验证方$\mathcal{V}$从可信第三方$\mathcal{I}$接收到结果，则接受或拒绝。

1.2.2.2 拉格朗日的选择多项式

上述多项式协议中，H为乘法子群，阶为N，生成元为g。对$i\in[N]$，令$L_i\in\mathbb{F}_{<N}[X]$为H的第i个拉格朗日多项式，且满足

$$L_i(g^j)=1, i=j$$
$$L_i(g^j)=0, i\neq j$$

对于$x\in H$，要求$L_i(x)f(x)=0$，则等价于$f(g^i)=0$。
因此，在H范围内进行点检测，拉格朗日多项式是非常有用的。

1.2.2.3 多项式与集合划分

乘法群上实现多项式

对整数n,d，向量$f\in\mathbb{F}^n,t\in\mathbb{F}^d$。使用符号$f\subset t$代替$\{f_i\}_{i\in[n]}\subset\{t_i\}_{i\in[n]}$。

令$H=\{g,\dots,g^{n+1}=1\}$为在域上的阶为$n+1$的乘法子群。
对于保密多项式$f\in\mathbb{F}[X]$且$i\in[n+1]$，记为$f_i:=f(g^i)$。
对于保密向量$f\in\mathbb{F}^n$，在域$\mathbb{F}_{<n}[X]$上的f保密多项式也记为$f(g^i)=f_i$。

当$f\subset t$，如果元素出现在f中的顺序与出现在t中的顺序相同，则称f由t划分。

对任意$i<i'\in[n]$，使得$f_i\neq f_{i'}$；如果$j,j'\in[d]$使得$t_j=f_i,t_{j'}=f_{i'}$，则$j<j'$。

给定$f\in\mathbb{F}^n,t\in\mathbb{F}^d,s\in\mathbb{F}^{n+d}$，如下定义 2 个二元多项式$F(\beta,\gamma)$和$G(\beta,\gamma)$

$$F(\beta,\gamma)\coloneqq(1+\beta)^n\cdot\prod_{i\in[n]}(\gamma+f_i)\prod_{i\in[d-1]}(\gamma(1+\beta)+t_i+\beta\cdot t_{i+1})$$

$$G(\beta,\gamma):=\prod_{i\in[n+d-1]}(\gamma(1+\beta)+s_i+\beta\cdot s_{i+1})$$

关键结论：二元多项式$F\equiv G$ 当且仅当 $f\subset t$且$s=(f,t)$由t划分。

证明：对F和G进行如下变换

$$F(\beta,\gamma):=(1+\beta)^{n+d-1}\cdot\prod_{i\in[n]}(\gamma+f_i)\prod_{i\in[d-1]}(\gamma+(t_i+\beta\cdot t_{i+1})/(1+\beta))$$
$$G(\beta,\gamma):=(1+\beta)^{n+d-1}\cdot\prod_{i\in[n+d-1]}(\gamma+(s_i+\beta\cdot s_{i+1})/(1+\beta))$$

情况 1： 如果$f\subset t$且$s=(f,t)$由t划分：
（1）对于每个$j\in[d-1]$，存在一个索引$i\in[n+d-1]$使得
$$(t_j,t_{j+1})=(s_j,s_{j+1})$$
即F和G中对应的因子是相等的
$$\gamma+(t_i+\beta\cdot t_{i+1})/(1+\beta)=\gamma+(s_i+\beta\cdot s_{i+1})/(1+\beta)$$
（2）令索引为i的$d-1$个元素记为集合$P'\subset[n+d-1]$，且补集$P:=[n+d-1]/P'$，则剩余的n个索引$i\in P$使得$s_i=s_{i+1}$且集合$\{s_i\}_{i\in P}$等于集合$\{f_i\}_{i\in[n]}$，即存在一个一一映射$j:P\to[n]$使得对于每个$i\in P$，有$s_i=f_{j(i)}$成立。

对于每个$i\in P$，对应的G的因子进行化简：$\gamma+(s_i+\beta s_{i+1})/(1+\beta)=\gamma+s_i$。

该因子等于F中的因子为$\gamma+f_{j(i)}$ 。因此，
$$\gamma+s_i=\gamma+f_{j(i)}$$
所以$F\equiv G$成立。

情况 2： 如果$F\equiv G$：
（1）对于每个$i\in[d-1]$，G一定包含一个等于$\gamma+(t_i+\beta t_{i+1})/(1+\beta)$的因子，即
$$\gamma+(t_i+\beta\cdot t_{i+1})/(1+\beta)=\gamma+(s_i+\beta\cdot s_{i+1})/(1+\beta)$$
上式表明$t_i+\beta t_{i+1}=s_i+\beta s_{i+1}$，根据 Schwartz–Zippel 引理得出$t_i=s_i,t_{i+1}=s_{i+1}$。
（2）令$P'\subset[n+d-1]$为索引为j的$d-1$个元素集合。对补集$j\in[n+d-1]/P'$，在F多项式中存在一个来自f的因子等于对应的G中的因子，即对于这类$j\in[n+d-1]/P'$，存在$i\in[n]$使得
$$\gamma+f_i=\gamma+(s_i+\beta\cdot s_{i+1})/(1+\beta)$$
则表明$f_i=s_i=s_{i+1}$。因此，有$f\subset t$且$s=(f,t)$由t划分。

综合情况 1 和情况 2，冲要条件成立。

1.2.3 Plookup 原理

预计算公开且正确的 Table 多项式：$t\in\mathbb{F}_{<n+1}[X]$，查找表由多项式值表达。
证明方$\mathcal{P}$输入保密多项式：$f\in\mathbb{F}_{<n}[X]$，**witness 由多项式值表达。**
1. 令$s\in\mathbb{F}^{2n+1}$为多项式f和t的组合$s=(f,t)$，由t划分。使用两个多项式$h_1,h_2\in\mathbb{F}_{<n+1}[X]$表达$s$多项式

$$h_1(g^i) = s_i, i \in [n+1],$$
$$h_2(g^i) = s_{n+i}, i \in [n+1]$$

2. 证明方$\mathcal{P}$计算多项式h_1, h_2并发送给可信第三方$\mathcal{I}$；（$\mathcal{P}$对 s 或h_1, h_2多项式承诺）
3. 验证方$\mathcal{V}$选择随机数$\beta, \gamma \in \mathbb{F}$并发送给证明方$\mathcal{P}$；（$\mathcal{P}$计算哈希值）
4. 证明方$\mathcal{P}$计算多项式$Z \in \mathbb{F}_{<n+1}[X]$，二元多项式聚合$F(\beta,\gamma)/G(\beta,\gamma)$函数值，

   (1) $Z(g) = 1,$

   (2) $Z(g^i) = \dfrac{(1+\beta)^{i-1} \cdot \prod_{j<i}(\gamma + f_j) \cdot \prod_{1<j<i}(\gamma(1+\beta) + t_j + \beta \cdot t_{j+1})}{\prod_{1<j<i}(\gamma(1+\beta) + s_j + \beta \cdot s_{j+1})(\gamma(1+\beta) + s_{n+j} + \beta \cdot s_{n+j+1})},$

   (3) $Z(g^{n+1}) = 1$

   证明方$\mathcal{P}$发送Z给可信第三方$\mathcal{I}$；（$\mathcal{P}$商多项式承诺）
5. 验证方$\mathcal{V}$检测Z成功，即$Z(g^{n+1}) = 1$。具体而言，对$x \in H$，验证方$\mathcal{V}$进行以下一致性验证：

   (1) $L_1(x)(Z(x) - 1) = 0,$

   (2) $(x - g^{n+1})Z(x)(1+\beta) \cdot (\gamma + f(x))(\gamma(1+\beta) + t(x) + \beta \cdot t(g \cdot x))$
   $= (x - g^{n+1})Z(g \cdot x)(\gamma(1+\beta) + h_1(x) + \beta \cdot h_1(g \cdot x))(\gamma(1+\beta) + h_2(x) + \beta \cdot h_2(g \cdot x)),$

   (3) $L_{n+1}(x)(h_1(x) - h_2(g \cdot x)) = 0,$

   (4) $L_{n+1}(x)(Z(x) - 1) = 0$

如果以上 4 式均成立，则接受，否则拒绝。

如果$\{t(g^i)\}_{i\in[n+1]} \not\supseteq \{f(g^i)\}_{i\in[n]}$，则在上述协议中，攻击者$\mathcal{A}$以证明方$\mathcal{P}$的角色运行协议，验证方$\mathcal{V}$接受的概率可忽略。此外，上述协议的证明长度为$5n+4$。

证明：攻击者$\mathcal{A}$作为证明方$\mathcal{P}$构造三个多项式$h_1, h_2, Z \in \mathbb{F}_{<n+1}[X]$证明发送给验证方$\mathcal{V}$，这三个多项式长为$3n+3$。验证方$\mathcal{V}$进行的第二项等式的一致性检测具有最高的阶，包含这三个多项式h_1, h_2, Z与线性多项项$X - g^{n+1}$的乘积。其中，f有n个元素，t有$n+1$个元素，Z有$n+1$个元素，可抵消多项式H有$n+1$个元素，所以商多项式为$2n+1$。因此，证明长度为$(3n+3)+(2n+1) = 5n+4$。

第 1 条等式确保$Z(g) = 1$；
第 4 条等式确保$Z(g^{n+1}) = 1$；
第 3 条等式表明$h_1(g^{n+1}) = h_2(g)$；
因此，h_1, h_2一致的描述了**单一向量**$s \in \mathbb{F}^{2n+1}$。

基于上节的**充要条件**，对于任意选择的 s，如果集合$\{t(g^i)\}_{i\in[n+1]}$不包含集合$\{f(g^i)\}_{i\in[n]}$，则多项式$F(X,Y)$与多项式$G(X,Y)$不同。基于 Schwartz–Zippel 引理，除去可忽略概率，随机选择β, γ会使得$F(\beta,\gamma) \neq G(\beta,\gamma)$，则$Z(g^{n+1}) \neq 1$。因此，验证方$\mathcal{V}$接受的概率可忽略。

反之，如果验证方$\mathcal{V}$接受，则第 4 条检测等式确保$Z(g^{n+1}) = 1$，结合第 1/2 条

$$
\begin{aligned}
Z(g^{n+1}) &= Z(g^n)\frac{(1+\beta)\cdot(\gamma+f(g^n))(\gamma(1+\beta)+t(g^n)+\beta t(g^{n+1}))}{(\gamma(1+\beta)+h_1(g^n)+\beta h_1(g^{n+1}))(\gamma(1+\beta)+h_2(g^n)+\beta h_2(g^{n+1}))} \\
&= Z(g^{n-1})\frac{(1+\beta)^2\cdot\prod_{j=n-1,n}(\gamma+f(g^j))\prod_{j=n-1,n}(\gamma(1+\beta)+t(g^j)+\beta t(g^{j+1}))}{\prod_{j=n-1,n}(\gamma(1+\beta)+h_1(g^n)+\beta h_1(g^{n+1}))(\gamma(1+\beta)+h_2(g^n)+\beta h_2(g^{n+1}))} \\
&= Z(g^{n-2})\ldots \\
&= Z(g)\frac{(1+\beta)^n\cdot\prod_{j<n+1}(\gamma+f_j)\cdot\prod_{1<j<n+1}(\gamma(1+\beta)+t_j+\beta t_{j+1})=\mathrm{F}}{\prod_{1<j<n+1}(\gamma(1+\beta)+s_j+\beta s_{j+1})(\gamma(1+\beta)+s_{n+j}+\beta s_{n+j+1})=G} \\
&= 1
\end{aligned}
$$

因此，$F \equiv G$，则推导出$f \subset t$。因此，证明方$\mathcal{P}$的 witness 在公开且正确的数据表中，则输入和输出的约束是正确的，则证明方$\mathcal{P}$的计算是正确。

1.2.4 Plookup 扩展

如果有w个多项式$f_1,\ldots,f_w \in \mathbb{F}_{<n}[X]$与一个对应的**公开且正确的查找表**$t^* \in (\mathbb{F}^w)^d$，需要检测对于每个$j \in [n]$，有$f_1(g^j),\ldots,f_w(g^j) \in t^*$。可使用**随机数**将多个多项式**线性组合**归约为一个多项式。

对于每个$i \in [w]$，**预计算公开且正确的表格**多项式为$t_i \in \mathbb{F}_{<d}[X]$。其中，$t_i(g^i) = t^*_{i,j}, j \in [d]$。

验证方$\mathcal{V}$选择随机数$\alpha \in \mathbb{F}$；（$\mathcal{P}$计算随机数）

预计算公开且正确的多项式$t := \sum_{i\in[w]} \alpha^i \cdot t_i$；

证明保密的输入多项式$f := \sum_{i\in[w]} \alpha^i \cdot f_i$。

Schwartz–Zippel 引理表明，除去可忽略概率，对于$j \in [n], f_1(g^j),\ldots,f_w(g^j) \notin t^*$，则$f(g^i) \notin t$。因此，可将多个多项式通过随机数$\alpha$进行组合，并运行上一节的协议实现查找表协议。

1.2.5 Plookup 范围证明

如果证明方$\mathcal{P}$想要证明$f \subset \{0,\ldots,d-1\}$，其中$d < n$。对于$i \in [d]$，可令$t_i = i - 1$实现检测。通过令$d = n+1$，使用**累加器/累乘器**则能够检测

$$f \subset \{0,\ldots,d-1\}$$

证明复杂度为$5n+4$。

此外，证明方$\mathcal{P}$能够证明$f \subset \{0,\ldots,2n-2\}$且验证方$\mathcal{V}$能够进行一致性检测。协议能够进行一般化扩展

$$f \subset \{0,\ldots,c(n-1)\}$$

该证明与检测而仅需要增加验证方$\mathcal{V}$的约束数量。因此，根据协议其他部分的最大约束程度，任意用户均可选择一个大数c，使得上述范围证明协议是一个子程序。

在域$\mathbb{F}$范围内，对s进行划分，从0到$c(n-1)$，即$s_1 = 0,\ldots,s_{2n} = c\cdot(n-1)$。对于每个$i \in [2n]$，均有$s_{i+1} - s_i \le c$。因此，推测出：对于每个$i \in [2n+1]$，$s_i \in \{0,\ldots,c\cdot(d-1)\}$。

通过一个约束条件$s_{i+1}-s_i \le c$，将差值带入一个能够整除$\{0,\dots,c\}$的阶为$c+1$的多项式。该约束增量能够排除差分之间的置换，且能够直接检查(f,t)与s之间的置换关系。其中，t是c的倍数，即$t_i = c\cdot(i-1), i\in[n]$。

乘法循环子群H，阶为$n+1$，生成元为g。协议参数为一个正整数c，多项式

$$P(X) := \sum_{i=0}^{c} (X-i)$$

公开且正确的预计算多项式：对于每个$i\in[n]$，$t_i = c\cdot(i-1)$，预计算多项式为$t\in\mathbb{F}_{<n}[X]$。

证明方$\mathcal{P}$的保密输入多项式：$f\in\mathbb{F}_{<n}[X]$。

1. 令$s\in\mathbb{F}^{2n+1}$为多项式f和t的组合，由t划分。使用两个多项式$h_1,h_2\in\mathbb{F}_{<n+1}[X]$表达$s=(f,t)$多项式
   (1) $h_1(g^i)=s_i, i\in[n+1]$,
   (2) $h_2(g^i)=s_{n+i}, i\in[n]$,
   (3) $h_2(g^{n+1})=c(n-1)$

2. 证明方$\mathcal{P}$计算多项式h_1,h_2并发送给可信第三方$\mathcal{I}$；（$\mathcal{P}$多项式承诺）
3. 验证方$\mathcal{V}$选择随机数$\gamma\in\mathbb{F}$并发送给证明方$\mathcal{P}$；　（$\mathcal{P}$计算哈希值）
4. 证明方计算多项式$Z\in\mathbb{F}_{<n+1}[X]$，该多项式聚合$F(\beta,\gamma)/G(\beta,\gamma)$的函数值，
   (1) $Z(g)=1$,
   (2) $Z(g^i)=\dfrac{(1+\beta)^{i-1}\cdot\prod_{j<i}(\gamma+f_j)\cdot\prod_{1<j<i}(\gamma(1+\beta)+t_j+\beta t_{j+1})}{\prod_{1<j<i}(\gamma(1+\beta)+s_j+\beta s_{j+1})(\gamma(1+\beta)+s_{n+j}+\beta s_{n+j+1})}, 2\le i\le n,$
   (3) $Z(g^{n+1})=1$

   证明方$\mathcal{P}$发送Z给可信第三方$\mathcal{I}$；　（$\mathcal{P}$随机打开点与商多项式承诺）
5. 对$x\in H$，验证方$\mathcal{V}$进行以下一致性验证：
   (1) $L_1(x)(h_1(x))=0$,
   (2) $P(h_1(g\cdot x)-h_1(x))=0$,
   (3) $P(h_2(g\cdot x)-h_2(x))=0$,
   (4) $L_{n+1}(x)(h_1(x)-h_2(g\cdot x))=0$,
   (5) $L_{n+1}(x)(h_2(x))=c\cdot(n-1)$,
   (6) $L_1(x)(Z(x)-1)=0$,
   (7) $(x-g^{n+1})Z(x)(\gamma+f(x))(\gamma+t(x))=(x-g^{n+1})Z(g\cdot x)(\gamma+h_1(x))(\gamma+h_2(x))$,
   (8) $L_{n+1}(x)(Z(x)-1)=0$

如果以上 8 式均成立，则接受，否则拒绝。

对于正整数c，如果集合$\{0,\dots,c\cdot(n-1)\}$不包含集合$\{f(g^i)\}_{i\in[n]}$，则在上述协议中，任意攻击者$\mathcal{A}$以证明方$\mathcal{P}$的角色运行协议，验证方$\mathcal{V}$接受的概率可忽略。此外，对任意$c\ge 2$，上述协议的证明长度为$(3+c)\cdot n+2$。

证明：攻击者$\mathcal{A}$作为证明方$\mathcal{P}$构造三个多项式$h_1,h_2,Z\in\mathbb{F}_{<n+1}[X]$证明发送给验证方$\mathcal{V}$，这三个多项式长为$3n+3$。 对于$c\ge 2$，最高阶的约束为$P(h_1(g\cdot X)-h(X))=0$和$P(h_2(g\cdot x)-h_2(x))=0$。其阶为$n\cdot(c+1)$，可抵消多项式$H$有$n+$

1个元素，所以商多项式为$n\cdot(c+1)-(n+1)$。因此，证明长度为$(3n+3)+n\cdot(c+1)-(n+1)=(3+c)n+2$。
对于某个$i\in[n]$，$f_i\notin\{0,\ldots,c(n-1)\}$，

（1）**验证等式 1 至 5** 表明$\{h_1(g^i),h_2(g^i)\}_{i\in[n]}$均在范围$\{0,\ldots,c(n-1)\}$内。

定义多项式

$$F(X):=\prod_{i\in[n]}(X-f(g^i))(X-t(g^i))$$
$$G(X):=\prod_{i\in[n]}(X-h_1(g^i))(X-h_2(g^i))$$

（2）**验证等式 6 至 8** 表明$F(\gamma)=G(\gamma)$。对于某个$i\in[n]$，如果$f_i\notin\{0,\ldots,c(n-1)\}$，则$F(\gamma)\neq G(\gamma)$，所以验证方$\mathcal{V}$接受的概率可忽略。

反之，如果验证方$\mathcal{V}$接受，则第 8 条检测等式确保$Z(g^{n+1})=1$，结合第 6/7 条

$$\begin{aligned}Z(g^{n+1})&=Z(g^n)\frac{(\gamma+f(g^n))(\gamma+t(g^n))}{(\gamma+h_1(g^n))(\gamma+h_2(g^n))}\\&=Z(g^{n-1})\frac{(\gamma+f(g^n))(\gamma+t(g^n))}{(\gamma+h_1(g^n))(\gamma+h_2(g^n))}\frac{(\gamma+f(g^{n-1}))(\gamma+t(g^{n-1}))}{(\gamma+h_1(g^{n-1}))(\gamma+h_2(g^{n-1}))}\\&=Z(g^{n-2})\ldots\\&=Z(g)\frac{\prod_{i\in[n]}(\gamma-f(g^i))(\gamma-t(g^i))}{\prod_{i\in[n]}(\gamma-h_1(g^i))(\gamma-h_2(g^i))}\\&=\frac{F(\gamma)}{G(\gamma)}\\&=1\end{aligned}$$

因此，$F\equiv G$。结合冲要条件，则推导出$f\subset\{0,\ldots,c\cdot(n-1)\}$。

2 UltraPlonK

2.1 预备知识

符号说明

n 次单位根也称为模 n 原根

$$H=\{\omega,\omega^2,\ldots\omega^{n-1},\omega^n=1\}$$

拉格朗日多项式

$$L_i(X)=\frac{\omega^i(X^n-1)}{n(X-\omega^i)}$$

目标多项式

$$Z_H(X)=(X-\omega)\cdot\ldots\cdot(X-\omega^{n-1})\cdot(X-\omega^n)=X^n-1$$

公开且正确的 3 列数据表$\mathrm{T}_{1,i},\mathrm{T}_{2,i},\mathrm{T}_{3,i},i=1,\ldots,n$，数据表的线性组合为

$$t_i=\mathrm{T}_{1,i}+\varsigma\cdot\mathrm{T}_{2,i}+\varsigma^2\cdot\mathrm{T}_{3,i},i=1,\ldots,n$$

使用**累加器**证明

$$\{f_i\}_{i=1}^{n} \subset \{t_i\}_{i=1}^{d}$$

数据表公开且正确，则保密数据 f_i 正确，且运算关系正确。令

$$f_i = \begin{cases} a(\omega^i) + \varsigma \cdot b(\omega^i) + \varsigma \cdot c(\omega^i), \text{if-the-ith-gate-is-a-lookup-gate} \\ \mathrm{T}_{1,i} + \varsigma \cdot \mathrm{T}_{2,i} + \varsigma^2 \cdot \mathrm{T}_{3,i}, \text{otherwise} \end{cases}$$

查找表优化

令 $\vec{f} = (f_0, ..., f_{n-1}), \vec{t} = (t_0, ..., t_{n-1})$，

原来需要校验 2 个等式

$$Z(X\omega) = Z(X) \frac{(1+\delta)(\varepsilon + f(X))(\varepsilon(1+\delta) + t(X) + \delta t(X\omega))}{(\varepsilon(1+\delta) + h_1(X) + \delta h_1(X\omega))(\varepsilon(1+\delta) + h_2(X) + \delta h_2(X\omega))}$$

$$L_{n-1}(X)h_1(X) - L_0(X)h_2(X) = 0$$

构造**随机差分集合** $\Delta t = (\Delta t_0, ..., \Delta t_{n-1}), \Delta s = (\Delta s_0, ..., \Delta s_{n-1})$，其中

$$\Delta t_i = \begin{cases} t_i + \delta t_{i+1}, \text{for}(i \in \{0, ..., n-2\}) \\ t_i + \delta t_0, \text{for}(i = n-1) \end{cases}$$

$$\Delta s_i = \begin{cases} s_i + \delta s_{i+1}, \text{for}(i \in \{0, ..., n-2\}) \\ s_i + \delta s_0, \text{for}(i = n-1) \end{cases}$$

构造随机差分集合后，仅需要校验一个等式

$$Z(X\omega) = Z(X) \frac{(1+\delta)(\varepsilon + f(X))(\varepsilon(1+\delta) + t(X) + \delta t(X\omega))}{(\varepsilon(1+\delta) + h_1(X) + \delta h_2(X\omega))(\varepsilon(1+\delta) + h_2(X) + \delta h_1(X\omega))}$$

公开且正确的表格 Table 多项式 $t(X) \in \mathbb{F}_{<n}[X]$

$$t(X) = \sum_{i=1}^{n} t_i L_i(X)$$

额外的查找表选择多项式 $q_K(X) \in \mathbb{F}_{<n}[X]$

$$q_K(\omega^i) \begin{cases} 1, \text{if-the-ith-gate-is-a-lookup-gate} \\ 0, \text{otherwise} \end{cases}$$

电路门选择多项式 $q_M(X), q_L(X), q_R(X), q_O(X), q_C(X) \in \mathbb{F}_{<n}[X]$

线多项式（置换多项式）

$$S_{\sigma_1}(X)=\sum_{i=1}^{n}\sigma^*(i)L_i(X),$$
$$S_{\sigma_2}(X)=\sum_{i=1}^{n}\sigma^*(n+i)L_i(X),$$
$$S_{\sigma_3}(X)=\sum_{i=1}^{n}\sigma^*(2n+i)L_i(X)$$

需要证明的运算关系：

$$\begin{aligned}&for(i\in[b])\\&Q_{L_i}\cdot a_i+Q_{R_i}\cdot b_i+Q_{O_i}\cdot c_i+Q_{M_i}\cdot a_i\cdot b_i+Q_{C_i}=0\\&q_{K_i}(\omega_i+\varsigma\cdot\omega_{n+i}+\varsigma\cdot\omega_{2n+i}-f_i)=0\\&f_i\in\{t_1,\dots,t_n\}\end{aligned}$$

2.2 核心协议

符号表达：G_1,G_2 分别是群 $\mathbb{G}_1,\mathbb{G}_2$ 的生成元，χ χ

$$[g]_1=[g(\chi)]_1=[g(X)]\cdot G_1\in\mathbb{G}_1,$$
$$[g]_2=[g(\chi)]_2=[g(X)]\cdot G_2\in\mathbb{G}_2$$

系统初始化：

Plonk 门数为 n，

（1）基于有毒废料生成 **KZG 的 PK**：$(\chi\cdot[1]_1,\dots,\chi^{n+5}\cdot[1]_1)$

（2）公开且正确的表格多项式 $\mathrm{T}_{1,i},\mathrm{T}_{2,i},\mathrm{T}_{3,i},i=1,\dots,n$

（3）门选择多项式：

$$(q_{M_i},q_{A_i},q_{B_i},q_{C_i},q_{Const_i})_{i=1}^{n},\sigma(X)$$
$$q_M(X)=\sum\nolimits_{i=1}^{n}q_{M_i}L_i(X)$$
$$q_A(X)=\sum\nolimits_{i=1}^{n}q_{A_i}L_i(X)$$
$$q_B(X)=\sum\nolimits_{i=1}^{n}q_{B_i}L_i(X)$$
$$q_C(X)=\sum\nolimits_{i=1}^{n}q_{C_i}L_i(X)$$
$$q_{Const}(X)=\sum\nolimits_{i=1}^{n}q_{Const_i}L_i(X)$$

（4）线多项式（置换多项式）：

$$S_{\sigma_1}(X)=\sum\nolimits_{i=1}^{n}\sigma(i)L_i(X)$$
$$S_{\sigma_2}(X)=\sum\nolimits_{i=1}^{n}\sigma(n+i)L_i(X)$$
$$S_{\sigma_3}(X)=\sum\nolimits_{i=1}^{n}\sigma(2n+i)L_i(X)$$

共同构造 **UltraPlonk 的 PK**。

Public input: $l,(w_i)_{i\in[l]}$

Prover input: $(w_i)_{i\in[3n]}$， **witness**

Round 1: 【不变】 **【电路门管脚数据多项式承诺】**电路门的输入三个信号多项式

生成随机数 $b_1,\ldots,b_6\in\mathbb{F}$

$$a(X)=(b_1X+b_2)Z_H(X)+\sum_{i=1}^{n}w_iL_i(X)$$
$$b(X)=(b_3X+b_4)Z_H(X)+\sum_{i=1}^{n}w_{n+i}L_i(X)$$
$$c(X)=(b_5X+b_6)Z_H(X)+\sum_{i=1}^{n}w_{2n+i}L_i(X)$$

输出**管脚数据三个多项式承诺** $[a]_1,[b]_1,[c]_1$

注意：

Rollup 功能，可以去掉随机项，减少计算复杂度，**仅实现数据压缩、计算压缩的功能，缺少零知识。**

zkRollup 功能，则不能去掉随机项。

Zcash 要实现零知识，则不能去掉随机项。

新增部分

Round 2:

基于承诺计算随机数 $\varsigma\in\mathbb{F}_p$；

保密数据向量 $\vec{f}=(f_0,\ldots,f_{n-1})$

公开且正确的表格向量 $\vec{t}=(t_0,\ldots,t_{n-1})$ 表达如下：

$$f_i=\begin{cases}a(\omega^i)+\varsigma\cdot b(\omega^i)+\varsigma\cdot c(\omega^i),\text{if-the-ith-gate-is-a-lookup-gate}\\ \mathrm{T}_{1,i}+\varsigma\cdot\mathrm{T}_{2,i}+\varsigma^2\cdot\mathrm{T}_{3,i},\text{otherwise}\end{cases}$$
$$t_i=\mathrm{T}_{1,i}+\varsigma\cdot\mathrm{T}_{2,i}+\varsigma^2\cdot\mathrm{T}_{3,i},i=1,\ldots,n$$

生成随机数 $b_7,\ldots,b_{13}\in\mathbb{F}$

计算**保密数据多项式** $f(X)$ **和公开且正确的表格数据多项式** $t(X)$

$$f(X)=(b_7X+b_8)Z_H(X)+\sum_{i=1}^{n}f_iL_i(X)$$
$$t(X)=\mathrm{T}_{1,i}(X)+\varsigma\cdot\mathrm{T}_{2,i}(X)+\varsigma^2\cdot\mathrm{T}_{3,i}(X)$$

令 $\vec{s}=(\vec{f},\vec{t})$ 由 $\vec{t}$ 划分，将 $\vec{s}$ 拆为 $\vec{h}_1,\vec{h}_2$，

$$\vec{h}_1=(s_1,s_3,...,s_{2n-1})$$
$$\vec{h}_2=(s_2,s_4,...,s_{2n})$$

计算对应的多项式

$$h_1(X)=(b_9X^2+b_{10}X+b_{11})Z_H(X)+\sum_{i=1}^{n}s_{2i-1}L_i(X)$$

$$h_2(X)=(b_{12}X^2+b_{13})Z_H(X)+\sum_{i=1}^{n}s_{2i}L_i(X)$$

输出三个多项式承诺 $[f(x)]_1,[h_1(x)]_1,[h_2(x)]_1$。

Round 3:【线数据多项式承诺】基于承诺计算随机数 $\beta,\gamma,\delta,\varepsilon\in\mathbb{F}_p$；

生成随机数 $b_{14},...,b_{19}\in\mathbb{F}$，

（1）计算置换多项式（线约束）

$$z_1(X)=(b_{14}X^2+b_{15}X+b_{16})Z_H(X)+$$
$$L_1(X)+$$
$$\sum_{i=1}^{n-1}L_{i+1}(X)\prod_{j=1}^{i}\frac{w_j+\beta\omega^{j-1}+\gamma}{w_j+\sigma(j)\beta+\gamma}\frac{w_{n+j}+\beta k_1\omega^{j-1}+\gamma}{w_{n+j}+\sigma(n+j)\beta+\gamma}\frac{w_{2n+j}+\beta k_2\omega^{j-1}+\gamma}{w_{2n+j}+\sigma(2n+j)\beta+\gamma}$$

输出**线数据多项式承诺** $[z]_1$

【公开且正确的 Table 多项式承诺】

$$z_2(X)=(b_{17}X^2+b_{18}X+b_{19})Z_H(X)+$$
$$\sum_{i=1}^{n-1}\left(L_{i+1}(X)\prod_{j=1}^{i}\frac{(1+\delta)(\varepsilon+f_j)(\varepsilon(1+\delta)+t_j+\delta t_{j+1})}{(\varepsilon(1+\delta)+s_{2j-1}+\delta s_{2j})(\varepsilon(1+\delta)+s_{2j}+\delta s_{2j+1})}\right)$$
$$\Leftrightarrow F\equiv G\Leftrightarrow f\subset t$$

输出**公开且正确的表格多项式承诺** $[z]_2$

添加零知识的**其他方法**，在**置换集合**中添加**随机数函数值**。

Round 4:【保密数据满足门约束与线约束、查找表】基于上述承诺计算随机数 $\alpha\in\mathbb{F}_p$。

商多项式存在，则确保上述门约束与线约束成立、查找表

$$q(X)=\frac{1}{Z_H(X)}\begin{cases}Gate:\left(q_A(X)\cdot a(X)+q_B(X)\cdot b(X)+q_C(X)\cdot c(X)+q_M(X)\cdot a(X)\cdot b(X)+q_{Const}(X)\right)\cdot 1\\ Line:+\left(\left(a(X)+\beta X+\gamma\right)\left(b(X)+\beta k_1X+\gamma\right)\left(c(X)+\beta k_2X+\gamma\right)z(X)\right)\cdot\alpha\\ Line:-\left(\left(a(X)+\beta S_{\sigma_1}(X)+\gamma\right)\left(b(X)+\beta k_1S_{\sigma_2}(X)+\gamma\right)\left(c(X)+\beta k_2S_{\sigma_3}(X)+\gamma\right)z(\omega X)\right)\cdot\alpha\\ Line:+\left(z(X)-1\right)L_1(X)\cdot\alpha^2\\ +q_K(X)(a(X)+\varsigma\cdot b(X)+\varsigma^2\cdot b(X)-f(X))\cdot\alpha^3\\ +z_2(X)(1+\delta)(\varepsilon+f(X))(\varepsilon(1+\delta)+t(X)+\delta t(X\omega))\cdot\alpha^4\\ -z_2(X\omega)(\varepsilon(1+\delta)+h_1(X)+\delta h_1(X\omega))(\varepsilon(1+\delta)+h_2(X)+\delta h_2(X\omega))\cdot\alpha^4\\ +(z_2(X)-1)L_1(X)\cdot\alpha^5\end{cases}$$

生成随机数$b_{20},\ldots,b_{22}\in\mathbb{F}$，

将$q(X)$分解为 3 个多项式

$$q(X)=\left(q_{low}(X)+b_{10}X^n\right)+\left(X^{n+1}q_{mid}(X)-b_{10}+b_{11}X^n\right)+\left(X^{2n+4}q_{high}(X)-b_{11}\right)$$

红色部分是随机项，缺少这部分，则无法实现零知识。

输出**这三个多项式承诺**$\left([q_{low}]_1,[q_{mid}]_1,[q_{high}]_1\right)$。

这个商多项式确保门约束和线约束、查找表约束正确。

Round 5: **【多项式随机打开点】**基于上述承诺计算随机数$\Im\in\mathbb{F}_p$。计算**多项式的值**

$$\begin{gathered}a(\Im),b(\Im),c(\Im),\\ S_{\sigma_1}(\Im),S_{\sigma_2}(\Im),\\ f(\Im),t(\Im),t(\omega\Im),\\ z_1(\omega\Im),z_2(\omega\Im),\\ h_1(\omega\Im),h_2(\omega\Im)\end{gathered}$$

输出上述**函数值**。

Round 6: 基于上述承诺计算随机数$v\in\mathbb{F}_p$。

计算一个辅助的**线性多项式**

$$r(X)=\begin{cases} Gate:\left(a(\Im)b(\Im)q_M(X)+a(\Im)q_A(X)+b(\Im)q_B(X)+c(\Im)q_C(X)+PI(\Im)+q_{Const}(X)\right)\cdot 1 \\ Line:+\alpha\cdot[\left(\left(a(\Im)+\beta\Im+\gamma\right)\left(b(\Im)+\beta k_1\Im+\gamma\right)\left(c(\Im)+\beta k_2\Im+\gamma\right)z_1(X)\right) \\ Line:-\left(\left(a(\Im)+\beta s_{\sigma_1}(\Im)+\gamma\right)\left(b(\Im)+\beta k_1 s_{\sigma_2}(\Im)+\gamma\right)\left(c(\Im)+\beta\cdot S_{\sigma_3}(\Im)+\gamma\right)\right)z_1(\Im\omega)] \\ Line:+\alpha^2\cdot(z_1(X)-1)L_1(\Im) \\ +\alpha^3\cdot q_K(X)(a(\Im)+\varsigma\cdot b(\Im)+\varsigma^2\cdot c(\Im)-f(\Im)) \\ +\alpha^4\cdot[z_2(X)(1+\delta)(\varepsilon+f(\Im))(\varepsilon(1+\delta)+t(\Im)+\delta t(\Im\omega)) \\ -z_2(\Im\omega)(\varepsilon(1+\delta)+h_1(\Im)+\delta h_1(\Im\omega))(\varepsilon(1+\delta)+h_2(\Im)+\delta h_2(\Im\omega))] \\ +\alpha^5\cdot(z_2(X)-1)L_1(\Im) \\ -\alpha^6\cdot Z_H(\Im)\left((q_{low}(\Im)+b_{10}\Im^n)+(\Im^{n+1}q_{mid}(\Im)-b_{10}+b_{11}\Im^n)+(\Im^{2n+4}q_{high}(\Im)-b_{11})\right) \end{cases}$$

计算 KZG 承诺中的商多项式，确保上述所有多项式正确

$$W_\Im(X)=\frac{1}{X-\Im}\begin{pmatrix} r(X) \\ +v\cdot\left(a(X)-a(\Im)\right) \\ +v^2\cdot\left(b(X)-b(\Im)\right) \\ +v^3\cdot\left(c(X)-c(\Im)\right) \\ +v^4\cdot\left(S_{\sigma_1}(X)-s_{\sigma_1}(\Im)\right) \\ +v^5\cdot\left(S_{\sigma_2}(X)-s_{\sigma_2}(\Im)\right) \\ +v^6\cdot\left(f(X)-f(\Im)\right) \\ +v^7\cdot\left(h_2(X)-h_2(\Im)\right) \\ +v^8\cdot\left(t(X)-t(\Im)\right) \end{pmatrix}$$

$$W_{\omega\Im}(X)=\frac{1}{X-\omega\Im}\begin{pmatrix} z_1(X)-z_1(\Im\omega) \\ +v\cdot\left(t(X)-t(\Im\omega)\right) \\ +v^2\cdot\left(z_2(X)-z_2(\Im\omega)\right) \\ +v^3\cdot\left(h_1(X)-h_1(\Im\omega)\right) \end{pmatrix}$$

计算并输出**商多项式**承诺$[W_\Im]_1,[W_{\omega\Im}]_1$。

商多项式存在，则确保多项式的随机打开点正确，包括：门多项式、线多项式、门约束与线性约束多项式、Table 多项式等均正确。

最终证明为

$$\pi_{SNARK}=\left(\begin{array}{l}Commit:[a(x)]_1,[b(x)]_1,[c(x)]_1,\\ Commit:[f(x)]_1,[h_1(x)]_1,[h_2(x)]_1,\\ Commit:[z_1(x)]_1,[z_2(x)]_1,\\ Commit:[q_{low}]_1,[q_{mid}]_1,[q_{high}]_1,\\ Commit:[W_{\mathfrak{I}}]_1,[W_{\omega\mathfrak{I}}]_1,\\ Value:a(\mathfrak{I}),b(\mathfrak{I}),c(\mathfrak{I}),\\ Value:s_{\sigma_1}(\mathfrak{I}),s_{\sigma_2}(\mathfrak{I}),\\ Value:f(\mathfrak{I}),t(\mathfrak{I}),t(\mathfrak{I}\omega),\\ Value:z_1(\mathfrak{I}\omega),z_2(\mathfrak{I}\omega),h_1(\mathfrak{I}\omega),h_2(\mathfrak{I})\end{array}\right)$$

验证密钥 VK 预先存储到以太坊一层合约

电路选择多项式承诺：$[q_M(x)]_1,[q_L(x)]_1,[q_R(x)]_1,[q_O(x)]_1,[q_C(x)]_1$

线约束多项式承诺：$[S_{\sigma_1}(x)]_1,[S_{\sigma_2}(x)]_1,[S_{\sigma_3}(x)]_1$

公开的数据表格多项式承诺：$[\mathrm{T}_{1,i}(x)]_1,[\mathrm{T}_{2,i}(x)]_1,[\mathrm{T}_{3,i}(x)]_1$

KZG 承诺的 VK：$[x]_2$

共同构成 **UltraPlonk** 的 **VK**。

验证算法

承诺值群元素合法性检查

$$\begin{array}{l}[a(x)]_1,[b(x)]_1,[c(x)]_1,\\ [f(x)]_1,[h_1(x)]_1,[h_2(x)]_1,\\ [z_1(x)]_1,[z_2(x)]_1,\\ [q_{low}]_1,[q_{mid}]_1,[q_{high}]_1,\\ [W_{\mathfrak{I}}]_1,[W_{\omega\mathfrak{I}}]_1\in\mathbb{G}_1\end{array}$$

函数值范围检测

$$\begin{array}{l}a(\mathfrak{I}),b(\mathfrak{I}),c(\mathfrak{I}),\\ s_{\sigma_1}(\mathfrak{I}),s_{\sigma_2}(\mathfrak{I}),\\ f(\mathfrak{I}),t(\mathfrak{I}),t(\omega\mathfrak{I}),\\ z_1(\mathfrak{I}\omega),z_2(\mathfrak{I}\omega),\\ h_1(\mathfrak{I}),h_2(\mathfrak{I})\in\mathbb{F}_p\end{array}$$

L 个公共输入范围检测 $(w_i)_{i\in[l]}\in\mathbb{F}_p^l$ 标量域

基于多项式承诺计算随机数$\varsigma,\beta,\gamma,\delta,\varepsilon,\theta,\alpha,\Im,v,\eta \in \mathbb{F}_p$

多项式求值$Z_H(\Im)=\Im^n-1$

拉格朗日基求值$L_1(\Im)=\dfrac{\omega(\Im^n-1)}{n(\Im-\omega)}$

公共输入多项式求值$PI(\Im)=\sum_{i\in l} w_i L_i(\Im)$

公开的数据表多项式承诺$[t(x)]_1=[\mathrm{T}_1(x)]_1+\varsigma\cdot[\mathrm{T}_2(x)]_1+\varsigma^2\cdot[\mathrm{T}_3(x)]_1$

计算一个辅助的线性多项式

$$
\begin{aligned}
& r_0 := PI(\Im) \\
& -\alpha\left(a(\Im)+\beta s_{\sigma_1}(\Im)+\gamma\right)\left(b(\Im)+\beta s_{\sigma_2}(\Im)+\gamma\right)\left(c(\Im)+\gamma\right)z_1(\omega\Im) \\
& -L_1(\Im)\alpha^2 \\
& -\alpha^4 z_2(\Im\omega)(\varepsilon(1+\delta)+\delta h_2(\Im))(\varepsilon(1+\delta)+h_1(\Im)+\delta h_1(\Im)) \\
& -\alpha^2\cdot L_1(\Im). \\
& r'(X) := r(X)-r_0
\end{aligned}
$$

基于多项式的值，计算多项式**函数值的承诺**

$$[D]_1=[r'(x)]_1+\eta\cdot[z_1(x)]_1+v^2\cdot[z_2(x)]_1+v^3\cdot[h_1(x)]_1$$

$$
\begin{aligned}
[D]_1 = & a(\Im)b(\Im)\,[q_M(x)]_1+a(\Im)\,[q_L(x)]_1+b(\Im)\,[q_R(x)]_1+c(\Im)\,[q_O(x)]_1+\,[q_C(x)]_1 \\
& +((a(\Im)\,+\beta\Im+\gamma)(b(\Im)\,+\beta k_1\Im+\gamma)(c(\Im)\,+\beta k_2\Im+\gamma)\alpha+L_1(\Im)\alpha^2+\eta)\,[z_1(x)]_1 \\
& -(a(\Im)+\beta S_{\sigma^1}(\Im)+\gamma)(b(\Im)+\beta S_{\sigma^2}(\Im)+\gamma)\alpha\beta z_1(\Im\omega)\,[S_{\sigma^3}(x)]_1 \\
& +(a(\Im)+\zeta b(\Im)+\zeta^2 c(\Im)-f(\Im))\alpha^3[q_K(x)]_1 \\
& +((1\,+\delta)(\varepsilon+f(\Im))(\varepsilon(1\,+\delta)+t(z)+\delta t(\Im\omega))\alpha^4+L_1(\Im)\alpha^5+\eta v^2)\,[z_2(x)]_1 \\
& +(\eta v^3-z_2(\Im\omega)(\varepsilon(1+\delta)+h_2(\Im)+\delta h1(\Im\omega))\alpha^4)\,[h_1(x)]_1 \\
& -Z_H(\Im)([q_{low}(x)]_1+\Im^{n+2}\cdot[q_{mid}(x)]_1+\Im^{n+4}\cdot[q_{high}(x)]_1)
\end{aligned}
$$

基于多项式函数值，计算多项式函数值的**承诺**的**线性组合**

$$
\begin{aligned}
[F]_1 = & [D]_1+v\cdot[a(x)]_1+v^2\cdot[b(x)]_1+v^3\cdot[c(x)]_1+v^4\cdot[S_{\sigma^1}(x)]_1+v^5\cdot[S_{\sigma^2}(x)]_1 \\
& +v^6\cdot[f(x)]_1+v^7\cdot[t(x)]1+v^8\cdot[h_2(x)]_1 \\
& +\eta([z_1(x)]_1+v\cdot[t(x)]_1+v^2\cdot[z_2(x)]_1+v^3\cdot[h_1(x)]_1)
\end{aligned}
$$

计算函数值的承诺

$$
[E]_1=\begin{bmatrix} -r_0+v\cdot a(\Im)+v^2\cdot b(\Im)+v^3\cdot c(\Im)+v^4\cdot s_{\sigma_1}(\Im)+v^5\cdot s_{\sigma_2}(\Im)+v^6\cdot f(\Im) \\ +v^7\cdot t(\Im)+v^8\cdot h_2(\Im)+u(z_1(\Im\omega)+v\cdot t(\Im\omega)+v^2\cdot z_2(\Im\omega)+v^3\cdot h_1(\Im\omega)) \end{bmatrix}_1
$$

双线性映射验证

$$e\left([W_\Im(x)]_1+u\cdot[W_{\omega\Im}(x)]_1,[\chi]_2\right)==e\left(\Im\cdot[W_\Im(x)]_1+u\Im\omega\cdot[W_{\omega\Im}(x)]_1+[F]_1-[E]_1,[1]_2\right)$$

lynndell 博士 新火科技 密码学专家 lynndell2010@gmail.com