ZK SHANGHAI

零知识证明工作坊

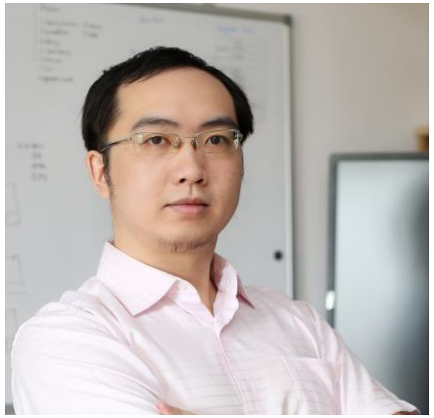
算术化

现代零知识密码学

Hosted by **SutuLabs** & **Kepler42B-ZK Planet**

课程资源：zkshanghai.xyz

个人介绍



梁爽

区块链 架构师

上海交大 计算机博士生
（休学创业中）

微信：icerdesign
微博：@wizicer
Github: @wizicer
Twitter: @icerdesign
LinkedIn: www.linkedin.com/in/icerdesign

1999年	• 正式开始学习写程序
2009年	• 在新媒传信（飞信）做高性能服务器程序架构及开发
2012年	• 在Honeywell工业控制部门做PLC、RTU上位机组态软件架构及开发
2017年	• 接触区块链，并开始创业开发区块链数据库
2020年	• 入学上海交大攻读博士学位，研究零知识证明数据库
2022年	• 获Chia全球开发大赛第一名，并开始Pawket钱包的开发
2023年	• 获得零知识链Mina的项目资助

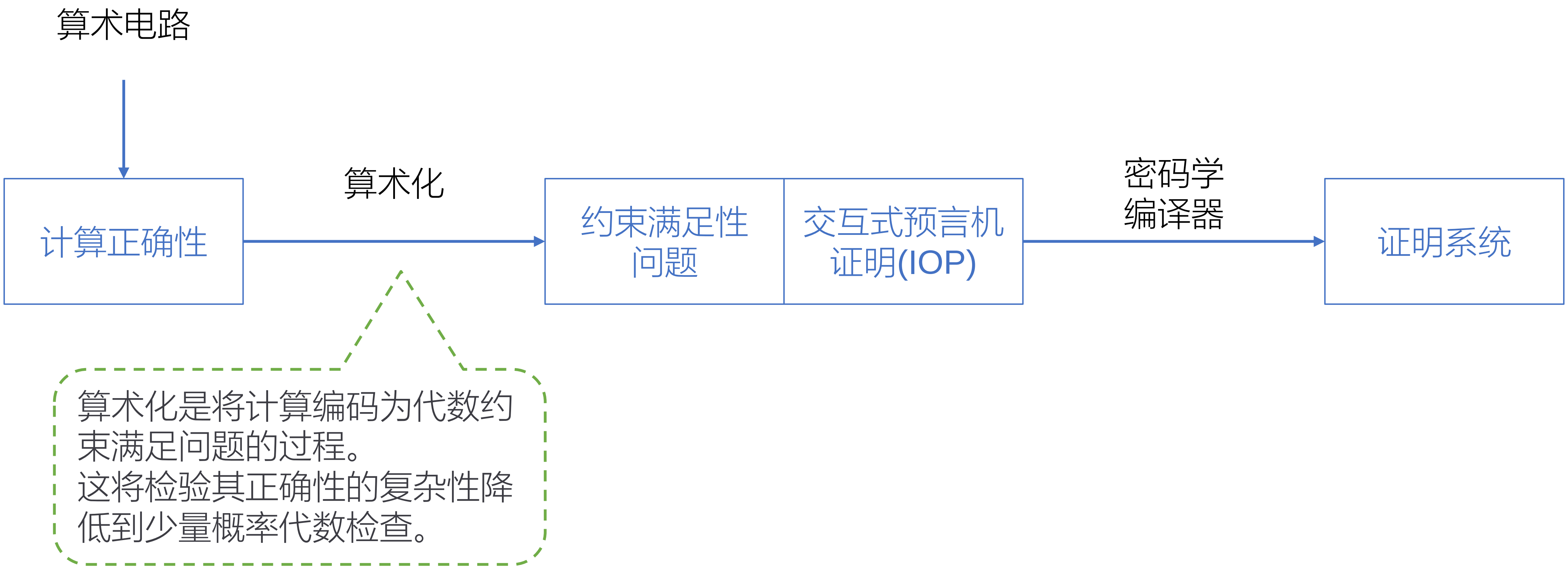
今日课程内容

- 模块化SNARK概述
- R1CS至QAP
  - 定义
  - 范例
- 代数中间表示AIR
  - 基本AIR
  - PAIR
  - RAP
  - 利用AIR构建虚拟机

今日课程将回答以下问题

- 电路如何变成多项式?
- 变成什么样的多项式?
- ZKVM的工作原理?

模块化SNARK


算术电路
计算正确性
算术化
约束满足性问题
交互式预言机证明(IOP)
密码学编译器
证明系统
算术化是将计算编码为代数约束满足问题的过程。
这将检验其正确性的复杂性降低到少量概率代数检查。

R1CS -> QAP

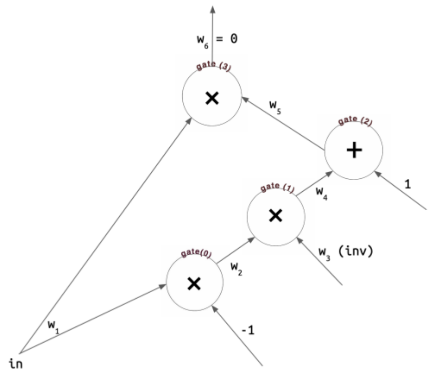
- 目标：将计算问题转换为单个多项式恒等式

算术电路

满足性问题

计算正确性 → 算术电路 → R1CS → QAP

```
template IsZero(){
    signal input in;
    signal output out;

    signal inv;

    inv <-- in != 0 ? 1 / in : 0;

    out <== -in * inv + 1;
    in * out === 0;
}
```

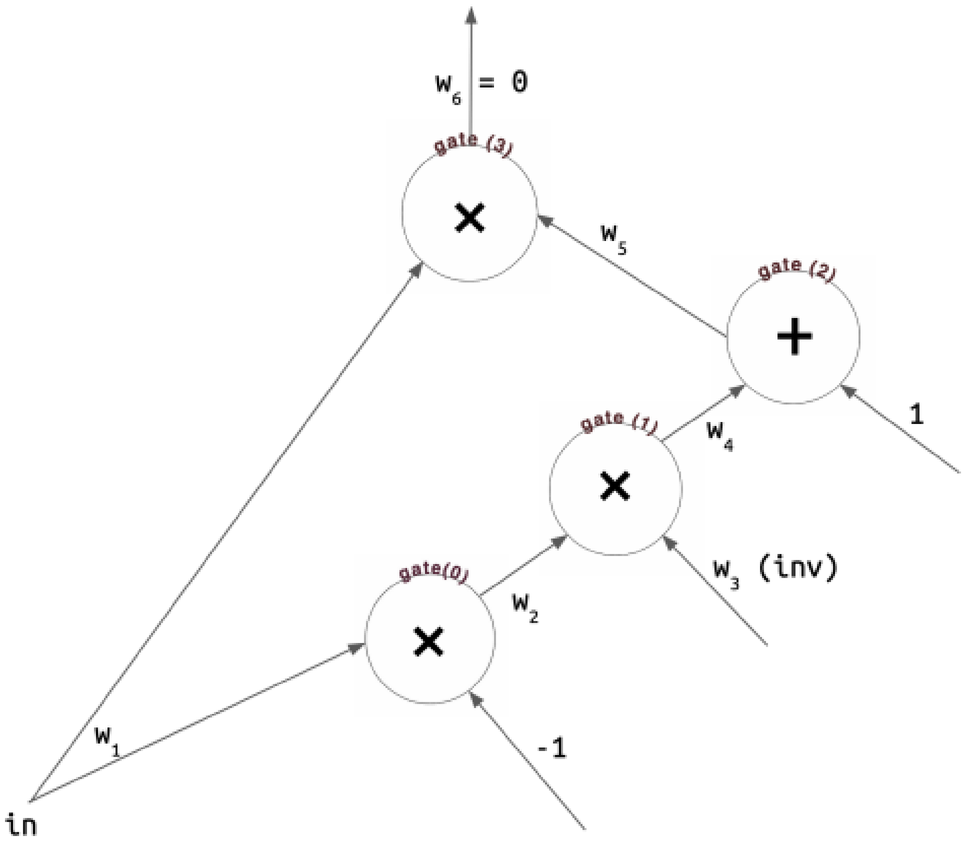


$$\mathcal{L}\vec{x} \cdot \mathcal{R}\vec{x} = \mathcal{O}\vec{x}$$

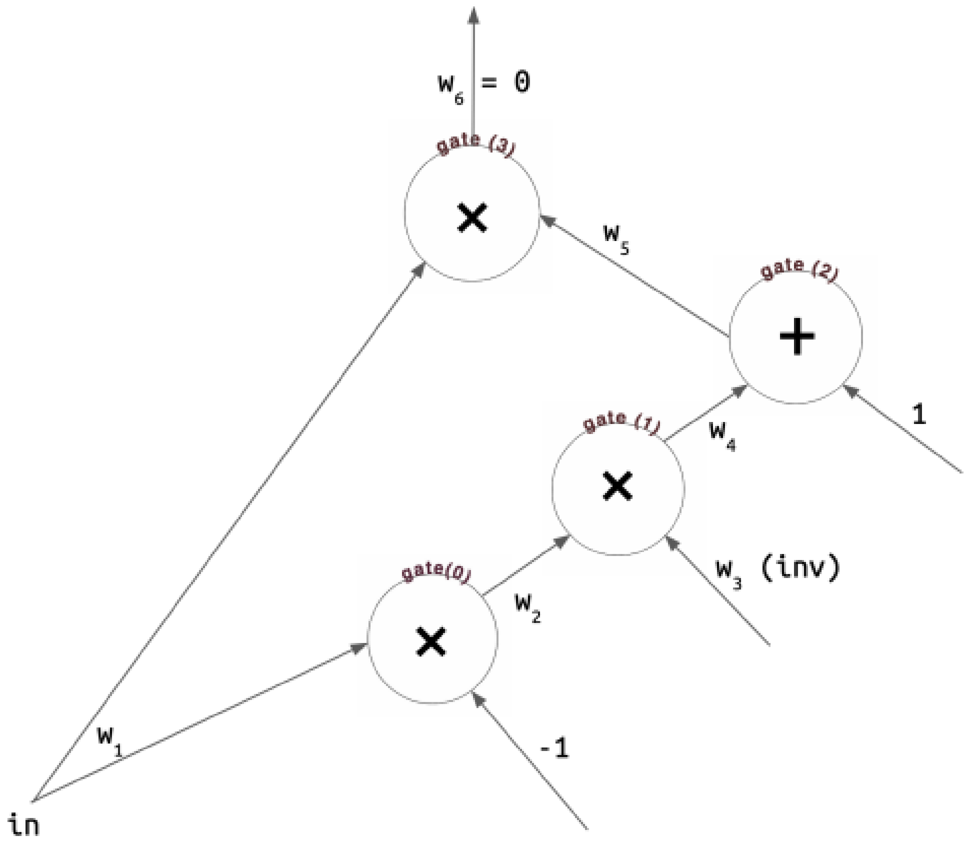
$$L(X) \cdot R(X) = O(X)$$

R1CS (IsZero)

```
template IsZero(){
    signal input in;
    signal output out;

    signal inv;

    inv <-- in != 0 ? 1 / in : 0;

    out <== -in * inv + 1;
    in * out === 0;
}
```



R1CS (IsZero)变平



- $g_0: w_1 \cdot (-1) = w_2$
- $g_1: w_2 \cdot w_3 = w_4$
- $g_2: (w_4 + 1) \cdot 1 = w_5$
- $g_3: w_1 \cdot w_5 = w_6$

R1CS (IsZero)

- $g_0: w_1 \cdot (-1) = w_2$
- $g_1: w_2 \cdot w_3 = w_4$
- $g_2: (w_4 + 1) \cdot 1 = w_5$
- $g_3: w_1 \cdot w_5 = w_6$

$$\begin{array}{ll} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \vec{l}_0 = & \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \vec{r}_0 = & \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \vec{o}_0 = & \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array},$$

$$\begin{array}{ll} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \vec{l}_1 = & \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \vec{r}_1 = & \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \\ \vec{o}_1 = & \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array},$$

$$\begin{array}{ll} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \vec{l}_2 = & \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \\ \vec{r}_2 = & \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \vec{o}_2 = & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array},$$

$$\begin{array}{ll} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \vec{l}_3 = & \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \vec{r}_3 = & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \\ \vec{o}_3 = & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}.$$

R1CS (IsZero)

$$\begin{array}{rccccccccc}
 & & w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & \\
\vec{l}_0 = & (& 0 & 1 & 0 & 0 & 0 & 0 & 0 &) \\
\vec{r}_0 = & (& -1 & 0 & 0 & 0 & 0 & 0 & 0 &), \\
\vec{o}_0 = & (& 0 & 0 & 1 & 0 & 0 & 0 & 0 &)
\end{array}$$

$$\begin{array}{rccccccccc}
 & & w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & \\
\vec{l}_1 = & (& 0 & 0 & 1 & 0 & 0 & 0 & 0 &) \\
\vec{r}_1 = & (& 0 & 0 & 0 & 1 & 0 & 0 & 0 &), \\
\vec{o}_1 = & (& 0 & 0 & 0 & 0 & 1 & 0 & 0 &)
\end{array}$$

$$\begin{array}{rccccccccc}
 & & w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & \\
\vec{l}_2 = & (& 1 & 0 & 0 & 0 & 1 & 0 & 0 &) \\
\vec{r}_2 = & (& 1 & 0 & 0 & 0 & 0 & 0 & 0 &), \\
\vec{o}_2 = & (& 0 & 0 & 0 & 0 & 0 & 1 & 0 &)
\end{array}$$

$$\begin{array}{rccccccccc}
 & & w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & \\
\vec{l}_3 = & (& 0 & 1 & 0 & 0 & 0 & 0 & 0 &) \\
\vec{r}_3 = & (& 0 & 0 & 0 & 0 & 0 & 1 & 0 &). \\
\vec{o}_3 = & (& 0 & 0 & 0 & 0 & 0 & 0 & 1 &)
\end{array}$$

R1CS

$$\mathcal{L}\vec{x} \cdot \mathcal{R}\vec{x} = \mathcal{O}\vec{x}$$
$$\vec{x} = (1, x_1, x_2, x_3, x_4, x_5, 0)$$

$$\mathcal{L} = \begin{array}{c} \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{array}{c} \\ \vec{l}_0 \\ \vec{l}_1 \\ \vec{l}_2 \\ \vec{l}_3 \end{array} ,$$

$$\mathcal{R} = \begin{array}{c} \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \begin{array}{c} \\ \vec{r}_0 \\ \vec{r}_1 \\ \vec{r}_2 \\ \vec{r}_3 \end{array} ,$$

$$\mathcal{O} = \begin{array}{c} \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array} \begin{array}{c} \\ \vec{o}_0 \\ \vec{o}_1 \\ \vec{o}_2 \\ \vec{o}_3 \end{array} .$$

R1CS -> QAP

$$\begin{array}{cc} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \mathcal{L} = & \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{array}{c} \vec{l}_0 \\ \vec{l}_1 \\ \vec{l}_2 \\ \vec{l}_3 \end{array},$$

$$\begin{array}{cc} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \mathcal{R} = & \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \begin{array}{c} \vec{r}_0 \\ \vec{r}_1 \\ \vec{r}_2 \\ \vec{r}_3 \end{array},$$

$$\begin{array}{cc} & \begin{array}{ccccccc} w_0 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 \end{array} \\ \mathcal{O} = & \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array} \begin{array}{c} \vec{o}_0 \\ \vec{o}_1 \\ \vec{o}_2 \\ \vec{o}_3 \end{array}.$$

- $L_j(i) = \mathcal{L}_{ij} = \vec{l_i}[j]$
- $R_j(i) = \mathcal{R}_{ij} = \vec{r_i}[j]$
- $O_j(i) = \mathcal{O}_{ij} = \vec{o_i}[j]$

- QAP判定式：
  $T(X) \mid P(X),$
  $P(X) \coloneqq L(X) \cdot R(X) - O(X)$
  $T(X) = \Pi(X - i)_{i=0}^{d-1}$

QAP

$$L_j(i) = \mathcal{L}_{ij}$$
$$X \in [0, d-1]$$

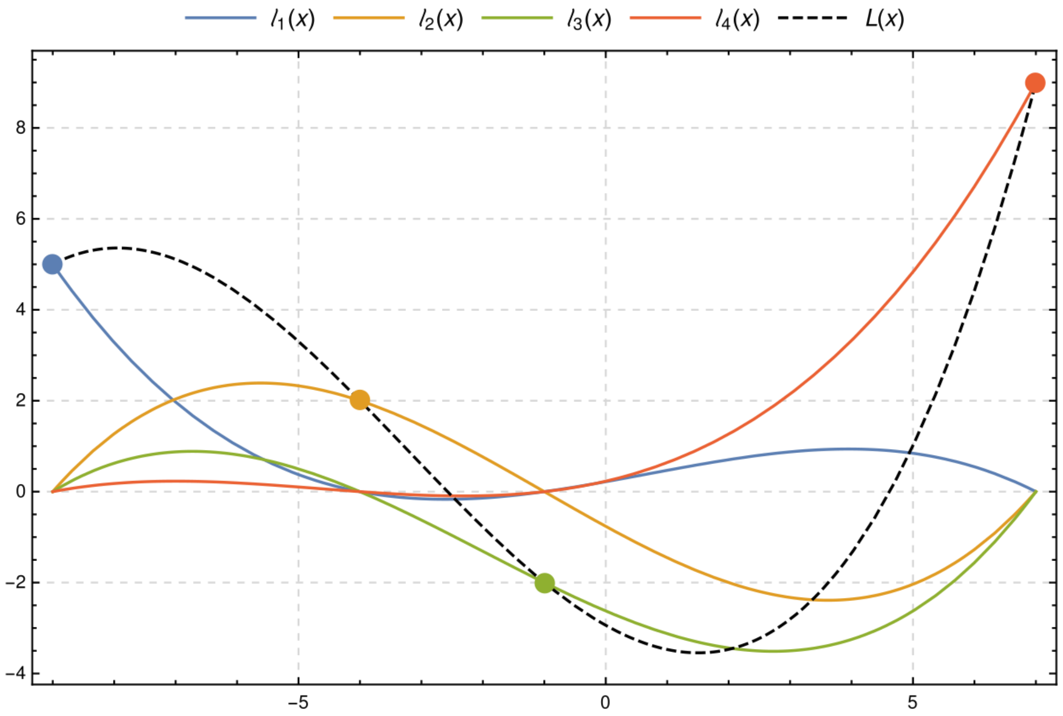
定义1.1. 二次算术程序(QAP)

一个度数为 d 、大小为 m 的二次算术程序 Q 由多项式 $\{L_j(X)\}, \{R_j(X)\}, \{O_j(X)\}, j \in [0, \ldots, m-1]$ 和一个目标多项式 $T(X) := \prod(X-i)_{i=0}^{d-1}$ 组成。当赋值 $(1, x_1, \ldots, x_{m-1})$ **满足** Q 时，

$$T(X) \mid P(X), P(X) := L(X) \cdot R(X) - O(X)$$

其中 $L(X) := \sum_{j=0}^{m-1} x_j \cdot L_j(X), R(X) := \sum_{j=0}^{m-1} x_j \cdot R_j(X), O(X) := \sum_{j=0}^{m-1} x_j \cdot O_j(X)$ 。

拉格朗日插值(Lagrange Interpolation)



- 该图显示了经过四个点$((-9,5),(-4,2),(-1,-2),(7,9))$的插值多项式$L(x)$（黑色虚线）
- 它是缩放基本多项式$y_0\ell_0(x)$，$y_1\ell_1(x)$，$y_2\ell_2(x)$和$y_3\ell_3(x)$的和。
- 插值多项式通过所有四个控制点，并且每个缩放基本多项式通过其相应的控制点，并且在x对应于其他三个控制点的位置为0。

拉格朗日插值(Lagrange Interpolation)

数学基础知识:拉格朗日插值

给定点和评估$\{(x_i, y_i)\}_{i=0}^{d-1}$，我们可以构造**一个插值多项式**$\mathcal{I}(X)$，使$\mathcal{I}(x_i) = y_i$:

$$\mathcal{I}(X) := \sum_{i=0}^{d-1} y_i \cdot \mathcal{L}_i(X)$$

其中，$\mathcal{L}_i(X)$是穿过评估值$\{x_0, \ldots, x_{d-1}\}$的拉格朗日基本多项式:

$$\mathcal{L}_i(X) := \prod_{x_j \neq x_i} \frac{X - x_j}{x_i - x_j} = \begin{cases} 1 \text{ if } X = x_i \\ 0 \text{ otherwise} \end{cases}$$

当评估域为$\{0, \ldots, d-1\}$时，当$X = i$，我们得到$\mathcal{L}_i(X) = 1$，否则为0。

当评估域为$\{\omega^0, \ldots, \omega^{n-1}\}$时，当$X = \omega^i$，我们得到$\mathcal{L}_i(X) = 1$，否则为0。

例子：R1CS转QAP

- $f(x) = x^3 + x + 5$

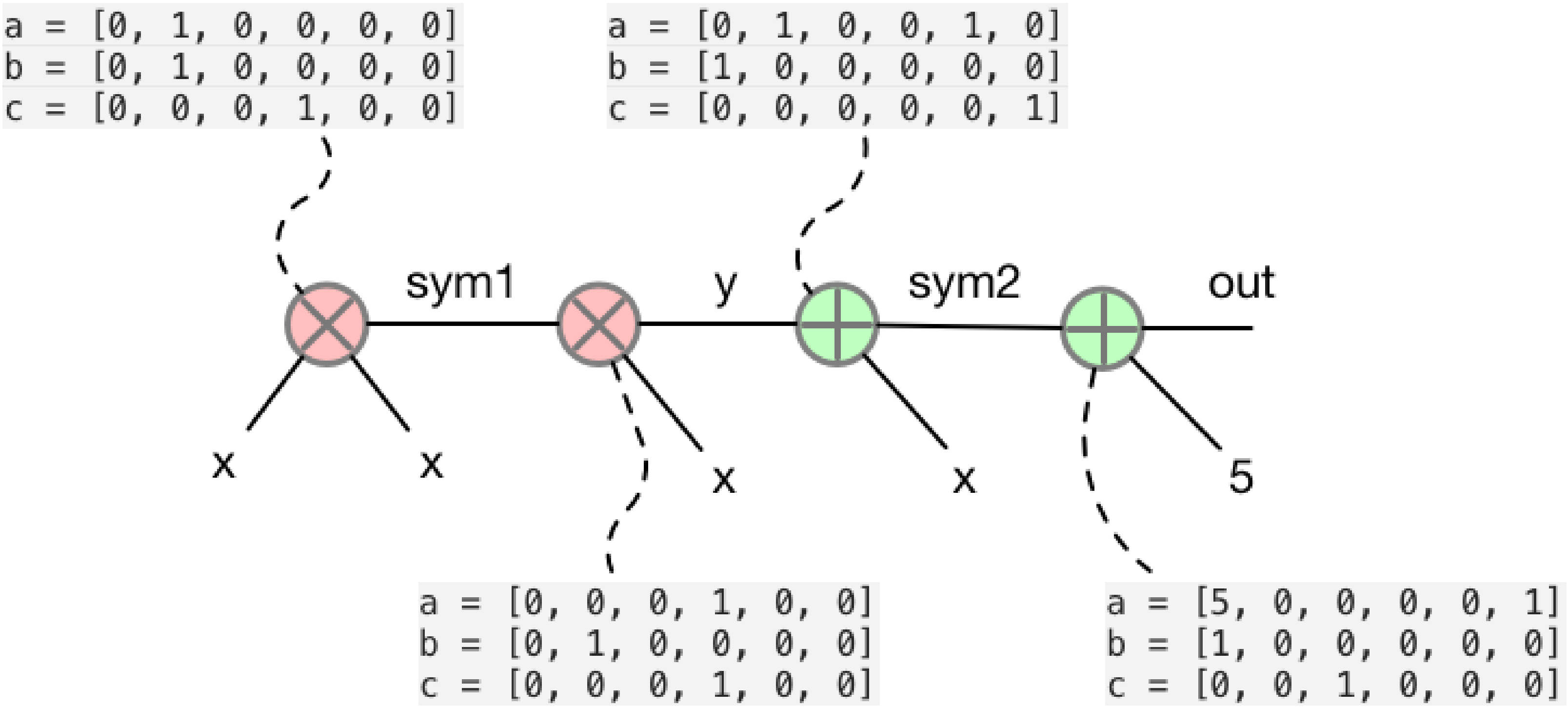
$$sym_1 = \underbrace{x \times x}_{x^2}$$

$$y = \underbrace{sym_1 \times x}_{x^3}$$

$$sym_2 = \underbrace{y + x}_{x^3+x}$$

$$out = \underbrace{sym_2 + 5}_{x^3+x+5}$$

$$s = [one, x, out, sym_1, y, sym_2]$$



	A	B	C
Gate1	[0, 1, 0, 0, 0, 0]	[0, 1, 0, 0, 0, 0]	[0, 0, 0, 1, 0, 0]
Gate2	[0, 0, 0, 1, 0, 0]	[0, 1, 0, 0, 0, 0]	[0, 0, 0, 0, 1, 0]
Gate3	[0, 1, 0, 0, 1, 0]	[1, 0, 0, 0, 0, 0]	[0, 0, 0, 0, 0, 1]
Gate4	[5, 0, 0, 0, 0, 1]	[1, 0, 0, 0, 0, 0]	[0, 0, 1, 0, 0, 0]

例子：R1CS转QAP

- $f(x) = x^3 + x + 5$
- 已知$x = 3$，算出中间值
- $s = [one, x, out, sym_1, y, sym_2]$
  $= [1,3,35,9,27,30]$

	A	B	C
Gate1	[0, 1, 0, 0, 0, 0]	[0, 1, 0, 0, 0, 0]	[0, 0, 0, 1, 0, 0]
Gate2	[0, 0, 0, 1, 0, 0]	[0, 1, 0, 0, 0, 0]	[0, 0, 0, 0, 1, 0]
Gate3	[0, 1, 0, 0, 1, 0]	[1, 0, 0, 0, 0, 0]	[0, 0, 0, 0, 0, 1]
Gate4	[5, 0, 0, 0, 0, 1]	[1, 0, 0, 0, 0, 0]	[0, 0, 1, 0, 0, 0]

利用拉格朗日插值，$A_1(x)$是找到的能通过$(1,0), (2,0), (3,0), (4,5)$的3阶多项式，
表示为$A_1(x) = 0.833x^3 - 5x^2 + 9.166x - 5$
QAP判定式：$T(X) \mid P(X), P(X) \coloneqq A(X) \cdot B(X) - C(X)$
其中：$T(X) = \Pi(X - i)_{i=0}^{d-1}$

x=1 x=2 x=3 x=4

	A	B	C
$A_1(x)$	[-5.0, 9.166, -5.0, 0.833]	[3.0, -5.166, 2.5, -0.333]	[0.0, 0.0, 0.0, 0.0]
$A_2(x)$	[8.0, -11.333, 5.0, -0.666]	[-2.0, 5.166, -2.5, 0.333]	[0.0, 0.0, 0.0, 0.0]
$A_3(x)$	[0.0, 0.0, 0.0, 0.0]	[0.0, 0.0, 0.0, 0.0]	[-1.0, 1.833, -1.0, 0.166]
$A_4(x)$	[-6.0, 9.5, -4.0, 0.5]	[0.0, 0.0, 0.0, 0.0]	[4.0, -4.333, 1.5, -0.166]
$A_5(x)$	[4.0, -7.0, 3.5, -0.5]	[0.0, 0.0, 0.0, 0.0]	[-6.0, 9.5, -4.0, 0.5]
$A_6(x)$	[-1.0, 1.833, -1.0, 0.166]	[0.0, 0.0, 0.0, 0.0]	[4.0, -7.0, 3.5, -0.5]

Fibonacci数列的AIR表示

$step$	a	b
$i=1$	1	1
$i=2$	2	3
$i=3$	5	8
$i=4$	13	21

- 转换程序：

$$f_1\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = A^{\text{next}} - (B + A);$$
$$f_2\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = B^{\text{next}} - \left(B + A^{\text{next}}\right).$$

- 范例：第i = 2行的状态转换

$$f_1\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = 5 - (3 + 2) = 0;$$
$$f_2\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = 8 - (5 + 3) = 0.$$

预处理的AIR（PAIR）

Preprocessed Algebraic Intermediate Representation

$step$	s_1	s_2	a	b
$i = 1$	1	0	0	1
$i = 2$	0	1	1	2
$i = 3$	1	1	2	2
$i = 4$	0	1	4	0

- 目标：同时启用加法和乘法
- 约束多项式为：

$$f\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = S_1 \cdot \left(A^{\text{next}} - (A + B)\right) + S_2 \cdot \left(A^{\text{next}} - A \cdot B\right).$$

- 在$i = 1$的行上检查约束：

$$f\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = 1 \cdot (1 - (0 + 1)) + 0 \cdot (1 - (0 \cdot 1)) = 0$$

- 在$i = 3$的行上检查约束：

$$f\left(X_1, X_2, X_1^{\text{next}}, X_2^{\text{next}}\right) = 1 \cdot (4 - (2 + 2)) + 1 \cdot (4 - (2 \cdot 2)) = 0$$

带预处理的随机化AIR（RAP）

Randomized AIR with Preprocessing

步骤	a	b	z
$i=1$	a_1	a_2	1
$i=2$	a_2	a_3	$\frac{(a_1+\gamma)}{(a_2+\gamma)}$
$i=3$	a_3	a_1	$\frac{(a_1+\gamma)(a_2+\gamma)}{(a_2+\gamma)(a_3+\gamma)}$
$i=4$	0	0	$\frac{(a_1+\gamma)(a_2+\gamma)(a_3+\gamma)}{(a_2+\gamma)(a_3+\gamma)(a_1+\gamma)}$

- 目标：多重集合相等性检查
- 约束多项式为：

$$\prod_{i\in[n]}(a_i+\gamma)=\prod_{i\in[n]}(b_i+\gamma)\Longrightarrow\prod_{i\in[n]}(a_i+\gamma)/(b_i+\gamma)=1$$

- 构建z列：

$$z_i=\prod_{1\le j\le i}(a_j+\gamma)/(b_j+\gamma)$$

- 检查约束：

$$Z^{\text{next}}\cdot(B+\gamma)-Z\cdot(A+\gamma)=0$$

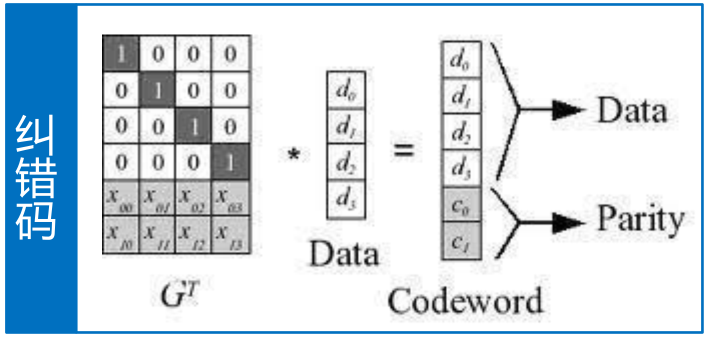
- 例如第$i=2$行：

$$\frac{(a_1+\gamma)(a_2+\gamma)}{(a_2+\gamma)(a_3+\gamma)}\cdot(a_3+\gamma)-\frac{(a_1+\gamma)}{(a_2+\gamma)}\cdot(a_2+\gamma)=0.$$

利用AIR构建虚拟机

算术化

- 算术化
  - 是将验证计算问题转换为检查某个多项式的问题，分两步：
  - 第一步：
    - 构建表格（执行踪迹）
    - 用多项式描述表格中各行/列间的数学关系
  - 第二步：（将这两个对象转换为一个低次多项式）
    - 利用纠错码将执行轨迹转为多项式
      - 哪怕仅一处错误的执行轨迹，会被纠错码放大，以至于与原执行轨迹几乎完全不同
    - 并扩展至更大的域
    - 用多项式约束将其转为低次多项式



算术化中的多项式变化

证明：我拥有512个数字，每个不是1就是0

(纠错码在本范例中忽略)

执行踪迹：

row
1
…
0

512

约束：$A_i \cdot A_i - A_i = 0$

$f(x)^2 - f(x) = 0$

执行踪迹转换为多项式
$\forall 0 \le i < 512: f(g^i) \coloneqq A_i$
定义f的根为$1, g, g^2, \ldots, g^{511}$

定义：
域：Z_{96769}（0到96768的正数域），
G定义为Z^*_{96769}（*为乘法群）的子群，
即$|G| = 512$（即该子群有512个元素），
g为G的生成元（群里的第1个元素）。

关于多项式及其根的一个基本事实是：
$p(x)$是多项式，某个特定值a使得$p(a) = 0$，
则一定存在多项式$q(x)$，当且仅当
$(x-a)q(x) = p(x), \deg(p) = \deg(q) + 1$。
因此，$\forall x \ne a, q(x) = \frac{p(x)}{(x-a)}$

对于k个根来说，设a_i是p的一个根，
$\forall i = 0 \ldots k-1$，存在一个阶为$\deg(p) - k$的多项式

$$q(x) = \frac{p(x)}{\prod_{i=0}^{k-1}(x - a_i)}$$

$$p(x) = \frac{f(x)^2 - f(x)}{\prod_{i=0}^{511}(x - g^i)}$$

$\forall x \notin \{1, g, g^2, \ldots, g^{511}\}$,
存在多项式q使得$\deg(q) = 2 * \deg(f) - 512$，
当且仅当执行踪迹是符合约束的。

低阶测试

Poly-IOP（FRI）

证据简洁及完美零知识性的关键

低阶测试(Low Degree Testing)：
是指通过仅对函数进行少量查询，来确定给定函数是否为某个有界阶数多项式的问题。
低阶测试已经研究了二十多年，是概率证明理论中的核心工具。

虚拟机状态转移算术化多项式

状态转移多项式约束

+
指令指针 增加1：$ip_{n+1} - ip_n - 1$
寄存器值 增加1：$reg_{n+1} - reg_n - 1$

-
指令指针 增加1：$ip_{n+1} - ip_n - 1$
寄存器值 减少1：$reg_{n+1} - reg_n + 1$

指令指针 ip	当前指令 ci	寄存器 reg
0	+	0
1	+	1
2	-	2
3	0	1

执行踪迹