Succinct Proofs

GKR

Rasmus Kirk Jakobsen

2026-01-19

Agenda

1. The GKR Protocol
2. Soundness
3. Efficiency
4. Need for Speed - A GKR-Based Grand Product Proof
5. The GKR-Based Grand Product Proof - Efficiency

The GKR Protocol

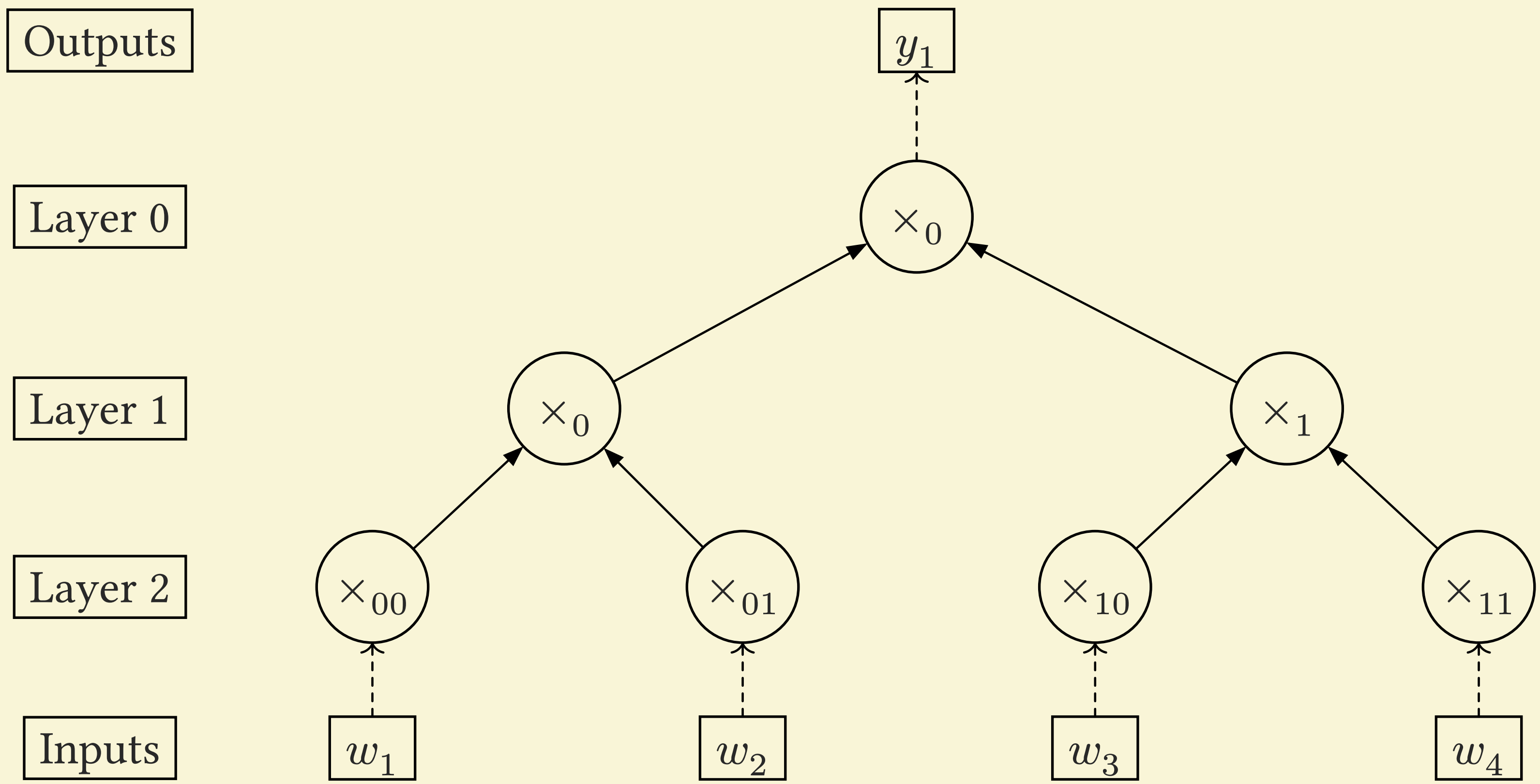
GKR Circuit

Layered arithmetic circuit $\mathcal{C}(\boldsymbol{w} \in \mathbb{F}^n) = \boldsymbol{y} \in \mathbb{F}^m$

- Only addition and multiplication gates
- n inputs
- m outputs,
- d layers

Example

$$S_0 = 1, S_1 = 2, S_2 = 4, s_i = \lg(S_i), n = 4, m = 1, d = 3$$



Polynomial Extension of W_i

$$W_i(\boldsymbol{a}) \in \mathbb{B}^{s_i} \to \mathbb{F}$$

$$\mathrm{add}_i(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \in \mathbb{B}^{s_i + 2s_{i+1}} \to \mathbb{B}$$

$$\mathrm{mul}_i(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \in \mathbb{B}^{s_i + 2s_{i+1}} \to \mathbb{B}$$

Polynomial Extension of W_i

$$W_i(\boldsymbol{a}) \in \mathbb{B}^{s_i} \to \mathbb{F}$$

$$\text{add}_i(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \in \mathbb{B}^{s_i + 2s_{i+1}} \to \mathbb{B}$$

$$\text{mul}_i(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \in \mathbb{B}^{s_i + 2s_{i+1}} \to \mathbb{B}$$

$$\widetilde{W}_i(\boldsymbol{a}) = \sum_{\boldsymbol{b}, \boldsymbol{c} \in \mathbb{B}^{s_{i+1}}} \widetilde{\text{add}}_i(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \Big(\widetilde{W}_{i+1}(\boldsymbol{b}) + \widetilde{W}_{i+1}(\boldsymbol{c}) \Big) + \widetilde{\text{mul}}_i(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{c})$$

Proving the Output of the Circuit

- Prover claims output of circuit is $\boldsymbol{y}' \cong W'$
- Actual output is $\boldsymbol{y} \cong W_0$

Proving the Output of the Circuit

- Prover claims output of circuit is $\boldsymbol{y}' \cong W'$
- Actual output is $\boldsymbol{y} \cong W_0$

$$\boldsymbol{r} \in_R \mathbb{F}^{s_0} : \widetilde{W}'(\boldsymbol{r}) = \widetilde{W}_0(\boldsymbol{r}) \Longrightarrow \widetilde{W}' = \widetilde{W}_0 \Longrightarrow W' = W_0$$

Sumcheck

$$\widetilde{W}_0(\boldsymbol{r}) = \sum_{\boldsymbol{b},\boldsymbol{c}\in\mathbb{B}^{s_1}} \widetilde{\mathrm{add}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c})\Big(\widetilde{W}_1(\boldsymbol{b}) + \widetilde{W}_1(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c}) \cdot \widetilde{W}_1(\boldsymbol{b}) \cdot \widetilde{W}_1(\boldsymbol{c})$$

$$\widetilde{W}_0(\boldsymbol{r}) = \sum_{\boldsymbol{b},\boldsymbol{c}\in\mathbb{B}^{s_1}} \widetilde{\mathrm{add}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c})\Big(\widetilde{W}_1(\boldsymbol{b}) + \widetilde{W}_1(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c}) \cdot \widetilde{W}_1(\boldsymbol{b}) \cdot \widetilde{W}_1(\boldsymbol{c})$$

Sumcheck polynomial

$$f_0(\boldsymbol{b},\boldsymbol{c}) = \widetilde{\mathrm{add}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c})\Big(\widetilde{W}_1(\boldsymbol{b}) + \widetilde{W}_1(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c}) \cdot \widetilde{W}_1(\boldsymbol{b}) \cdot \widetilde{W}_1(\boldsymbol{c})$$

$$\widetilde{W}_0(\boldsymbol{r}) = \sum_{\boldsymbol{b},\boldsymbol{c}\in\mathbb{B}^{s_1}} \widetilde{\mathrm{add}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c})\Big(\widetilde{W}_1(\boldsymbol{b}) + \widetilde{W}_1(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c}) \cdot \widetilde{W}_1(\boldsymbol{b}) \cdot \widetilde{W}_1(\boldsymbol{c})$$

Sumcheck polynomial

$$f_0(\boldsymbol{b},\boldsymbol{c}) = \widetilde{\mathrm{add}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c})\Big(\widetilde{W}_1(\boldsymbol{b}) + \widetilde{W}_1(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r},\boldsymbol{b},\boldsymbol{c}) \cdot \widetilde{W}_1(\boldsymbol{b}) \cdot \widetilde{W}_1(\boldsymbol{c})$$

Last round of sumcheck

$$\boldsymbol{b}_1', \boldsymbol{c}_1' \in_R \mathbb{F}^{s_{i+1}}$$

$$f_0(\boldsymbol{b}_1',\boldsymbol{c}_1') = \widetilde{\mathrm{add}}_0(\boldsymbol{r},\boldsymbol{b}_1',\boldsymbol{c}_1')\Big(\widetilde{W}_1(\boldsymbol{b}_1') + \widetilde{W}_1(\boldsymbol{c}_1')\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r}_0',\boldsymbol{b}_1',\boldsymbol{c}_1') \cdot \widetilde{W}_1(\boldsymbol{b}_1') \cdot \widetilde{W}_1(\boldsymbol{c}_1')$$

Idea: Verify Evaluations using Sumcheck

$$f_1(\boldsymbol{b}, \boldsymbol{c}) = \widetilde{\mathrm{add}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c})\Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_1(\boldsymbol{b}_1', \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c})$$

$$f_1(\boldsymbol{b}, \boldsymbol{c}) = \widetilde{\mathrm{add}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c})\Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_1(\boldsymbol{c}_1', \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c})$$

Exponential!

Combining two claims to one

$$
\begin{aligned}
q(\boldsymbol{b}_0', \boldsymbol{c}_0') &= \widetilde{W}_1(\boldsymbol{b}_0') + \alpha \cdot \widetilde{W}_1(\boldsymbol{c}_0') \\
&= \left(\sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^{s_2}} \widetilde{\mathrm{add}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) \Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c}) \Big) + \widetilde{\mathrm{mul}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c}) \right) + \\
&\quad\ \alpha \cdot \left(\sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^{s_2}} \widetilde{\mathrm{add}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c}) \Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c}) \Big) + \widetilde{\mathrm{mul}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c}) \right) \\
&= \sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^{s_2}} \widetilde{\mathrm{add}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) \Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c}) \Big) + \widetilde{\mathrm{mul}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c}) + \\
&\quad\ \alpha \cdot \widetilde{\mathrm{add}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c}) \Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c}) \Big) + \alpha \cdot \widetilde{\mathrm{mul}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c}) \\
&= \sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^{s_2}} \Big(\widetilde{\mathrm{add}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{add}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c}) \Big) \Big(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c}) \Big) + \\
&\quad\ \Big(\widetilde{\mathrm{mul}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{mul}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c}) \Big) \Big(\widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c}) \Big)
\end{aligned}
$$

Combining two claims to one

We have:

$$
\begin{aligned}
v_1 &:= p(\boldsymbol{r}_1), v_2 := p(\boldsymbol{r}_2) \\
q(X_1, .., X_{2\ell}) &:= p(X_1, ..., X_\ell) + \alpha \cdot p(X_{\ell+1}, ..., X_{2\ell} g) \\
q(\boldsymbol{r}_1, \boldsymbol{r}_2) &\stackrel{?}{=} v_1 + \alpha \cdot v_2
\end{aligned}
$$

Combining two claims to one

We have:

$$v_1 := p(\boldsymbol{r}_1), v_2 := p(\boldsymbol{r}_2)$$
$$q(X_1, .., X_{2\ell}) := p(X_1, ..., X_\ell) + \alpha \cdot p(X_{\ell+1}, ..., X_{2\ell} g)$$
$$q(\boldsymbol{r}_1, \boldsymbol{r}_2) \stackrel{?}{=} v_1 + \alpha \cdot v_2$$

Proof: Assume $v_1 \neq p(\boldsymbol{r}_1) \vee v_2 \neq p(\boldsymbol{r}_2)$, but q is defined as above:

$$e(X) = v_1 + X \cdot v_2 - (p(\boldsymbol{r}_1) + X \cdot p(\boldsymbol{r}_2))$$
$$\Pr[e(\alpha) = 0 \mid e(X) \neq 0] = \frac{\deg(e)}{|\mathbb{F}|} = \frac{1}{|\mathbb{F}|}$$

Combining two claims to one

The next round we sumcheck:

$$f_1(\boldsymbol{b}, \boldsymbol{c}) := \left(\widetilde{\mathrm{add}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{add}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c})\right)\left(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c})\right) + \\ \left(\widetilde{\mathrm{mul}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{mul}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c})\right)\left(\widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c})\right)$$

Combining two claims to one

The next round we sumcheck:

$$f_1(\boldsymbol{b}, \boldsymbol{c}) := \left(\widetilde{\mathrm{add}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{add}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c})\right)\left(\widetilde{W}_2(\boldsymbol{b}) + \widetilde{W}_2(\boldsymbol{c})\right) + \left(\widetilde{\mathrm{mul}}_1(\boldsymbol{b}_0', \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{mul}}_1(\boldsymbol{c}_0', \boldsymbol{b}, \boldsymbol{c})\right)\left(\widetilde{W}_2(\boldsymbol{b}) \cdot \widetilde{W}_2(\boldsymbol{c})\right)$$

And the verifier checks:

$$m_i \stackrel{?}{=} \left(\widetilde{\mathrm{add}}_i(\boldsymbol{b}_{i-1}', \boldsymbol{b}_i', \boldsymbol{c}_i') + \alpha \cdot \widetilde{\mathrm{add}}_i(\boldsymbol{c}_{i-1}', \boldsymbol{b}_i', \boldsymbol{c}_i')\right)\left(v_{\boldsymbol{b}_i'} + v_{\boldsymbol{c}_i'}\right) + \left(\widetilde{\mathrm{mul}}_i(\boldsymbol{b}_{i-1}', \boldsymbol{b}_i', \boldsymbol{c}_i') + \alpha \cdot \widetilde{\mathrm{mul}}_i(\boldsymbol{c}_{i-1}', \boldsymbol{b}_i', \boldsymbol{c}_i')\right)\left(v_{\boldsymbol{b}_i'} \cdot v_{\boldsymbol{c}_i'}\right)$$

Full Protocol: Preprocessing

$\mathcal{P}$ sends W' to $\mathcal{V}$ claiming that $W' = W_0$. $\mathcal{V}$ then picks out a random $\boldsymbol{r}$. After this point, $\mathcal{P}$ wants to prove that $m_0 = \widetilde{W}_0(\boldsymbol{r})$.

$$W' : \mathbb{B}^{s_0} \to \mathbb{F} \xrightarrow{\qquad\quad W' \quad\qquad} \boldsymbol{r} \in_R \mathbb{F}^{s_0}, m_0 := \widetilde{W}'(\boldsymbol{r})$$

Full Protocol: Round zero

$$f_0(\boldsymbol{b}, \boldsymbol{c}) = \widetilde{\mathrm{add}}_0(\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{c})\Big(\widetilde{W}_1(\boldsymbol{b}) + \widetilde{W}_1(\boldsymbol{c})\Big) + \widetilde{\mathrm{mul}}_0(\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{c}) \cdot \widetilde{W}_1(\boldsymbol{b}) \cdot \widetilde{W}_1(\boldsymbol{c})$$

$$\xleftrightarrow{\qquad \sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^{s_1}} f_0(\boldsymbol{b}, \boldsymbol{c}) \stackrel{?}{=} m_0 \qquad}$$

At the end of the protocol, $\mathcal{P}$ sends $\mathcal{V}$ the evaluations of $v_{\boldsymbol{b}'_0} := \widetilde{W}_1(\boldsymbol{b}'_0)$ and $v_{\boldsymbol{c}'_0} := \widetilde{W}_1(\boldsymbol{c}'_0)$, so $\mathcal{V}$ can make the final check in the sumcheck protocol.

$$v_{\boldsymbol{b}'_0} := \widetilde{W}_1(\boldsymbol{b}'_0), v_{\boldsymbol{c}'_0} := \widetilde{W}_1(\boldsymbol{c}'_0) \xrightarrow{\quad v_{\boldsymbol{b}'_0}, v_{\boldsymbol{c}'_0} \quad} \begin{array}{l} m_0 \stackrel{?}{=} \widetilde{\mathrm{add}}_0(\boldsymbol{r}, \boldsymbol{b}'_0, \boldsymbol{c}'_0) \cdot \left(v_{\boldsymbol{b}'_0} + v_{\boldsymbol{c}'_0}\right) + \\ \widetilde{\mathrm{mul}}_0(\boldsymbol{r}, \boldsymbol{b}'_0, \boldsymbol{c}'_0) \cdot v_{\boldsymbol{b}'_0} \cdot v_{\boldsymbol{c}'_0} \end{array}$$

Full Protocol: Round i

$$f_i(\boldsymbol{b}, \boldsymbol{c}) := \left(\widetilde{\mathrm{add}}_i(\boldsymbol{b}'_{i-1}, \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{add}}_i(\boldsymbol{c}'_{i-1}, \boldsymbol{b}, \boldsymbol{c})\right)\left(\widetilde{W}_{i+1}(\boldsymbol{b}) + \widetilde{W}_{i+1}(\boldsymbol{c})\right) + \left(\widetilde{\mathrm{mul}}_i(\boldsymbol{b}'_{i-1}, \boldsymbol{b}, \boldsymbol{c}) + \alpha \cdot \widetilde{\mathrm{mul}}_i(\boldsymbol{c}'_{i-1}, \boldsymbol{b}, \boldsymbol{c})\right)\left(\widetilde{W}_{i+1}(\boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{c})\right)$$

$$f_i(\boldsymbol{b}, \boldsymbol{c}) \xleftarrow{\quad \alpha \quad} \alpha \in_R \mathbb{F}$$

$$\xleftrightarrow{\sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^{s_{i+1}}} f_i(\boldsymbol{b}, \boldsymbol{c}) \stackrel{?}{=} m_i}$$

$$m'_i := \left(\widetilde{\mathrm{add}}_i(\boldsymbol{b}'_{i-1}, \boldsymbol{b}'_i, \boldsymbol{c}'_i) + \alpha \cdot \widetilde{\mathrm{add}}_i(\boldsymbol{c}'_{i-1}, \boldsymbol{b}'_i, \boldsymbol{c}'_i)\right)\left(v_{\boldsymbol{b}'_i} + v_{\boldsymbol{c}'_i}\right) + \left(\widetilde{\mathrm{mul}}_i(\boldsymbol{b}'_{i-1}, \boldsymbol{b}'_i, \boldsymbol{c}'_i) + \alpha \cdot \widetilde{\mathrm{mul}}_i(\boldsymbol{c}'_{i-1}, \boldsymbol{b}'_i, \boldsymbol{c}'_i)\right)\left(v_{\boldsymbol{b}'_i} \cdot v_{\boldsymbol{c}'_i}\right)$$

$$\begin{array}{l} v_{\boldsymbol{b}'_i} := \widetilde{W}_{i+1}(\boldsymbol{b}'_i), \\ v_{\boldsymbol{c}'_i} := \widetilde{W}_{i+1}(\boldsymbol{c}'_i) \end{array} \xrightarrow{\quad v_{\boldsymbol{b}'_i},\, v_{\boldsymbol{c}'_i} \quad} m_i \stackrel{?}{=} m'_i$$

At the input layer d, $\mathcal{V}$ has two claims $v_{\boldsymbol{b}'_{d-1}}$ and $v_{\boldsymbol{c}'_{d-1}}$. $\mathcal{V}$ constructs $\widetilde{W}_d$ from $\boldsymbol{w}$. $\mathcal{V}$ then finally checks that $\widetilde{W}_d(\boldsymbol{b}'_{d-1}) \stackrel{?}{=} v_{\boldsymbol{b}'_{d-1}}$ and $\widetilde{W}_d(\boldsymbol{c}'_{d-1}) \stackrel{?}{=} v_{\boldsymbol{c}'_{d-1}}$.

$$\widetilde{W}_d(\boldsymbol{b}'_{d-1}) \stackrel{?}{=} v_{\boldsymbol{b}'_{d-1}} \wedge$$
$$\widetilde{W}_d(\boldsymbol{c}'_{d-1}) \stackrel{?}{=} v_{\boldsymbol{c}'_{d-1}}$$

Soundness

Preprocessing:

- Check $\widetilde{W}' \stackrel{?}{=} \widetilde{W}_0$
- Soundness Error: $s_0/|\mathbb{F}|$

Preprocessing:

- Check $\widetilde{W}' \stackrel{?}{=} \widetilde{W}_0$
- Soundness Error: $s_0/|\mathbb{F}|$

Sumcheck Round j:

- Check $f_i^{(j)}\left(r_{j-1}\right) \stackrel{?}{=} f_i^{(j)}(0) + f_i^{(j)}(1)$
- Soundness Error: $\deg\left(f_i^{(j)}\right)/|\mathbb{F}| = 2/|\mathbb{F}|$

Preprocessing:

- Check $\widetilde{W}' \stackrel{?}{=} \widetilde{W}_0$
- Soundness Error: $s_0/|\mathbb{F}|$

Sumcheck Round j:

- Check $f_i^{(j)}\left(r_{j-1}\right) \stackrel{?}{=} f_i^{(j)}(0) + f_i^{(j)}(1)$
- Soundness Error: $\deg\left(f_i^{(j)}\right)/|\mathbb{F}| = 2/|\mathbb{F}|$

GKR Round i:

- Check: $m_i \stackrel{?}{=} m_i'$
- Soundness Error: $\frac{1}{|\mathbb{F}|}$

Soundness Bound

$$\begin{aligned}
\delta_s &= \Pr[E_{W'}] + \bigcup_{i=0}^{d-1} \bigcup_{j=0}^{s_i} \Pr\left[E_j\right] + \bigcup_{i=1}^{d-1} \Pr[E_i] \\
&\leq \frac{s_0}{|\mathbb{F}|} + \sum_{i=0}^{d-1} \sum_{j=0}^{s_i} \frac{2}{|\mathbb{F}|} + \sum_{i=1}^{d-1} \frac{1}{|\mathbb{F}|} \\
&= \frac{s_0}{|\mathbb{F}|} + \sum_{i=0}^{d-1} s_i \frac{2}{|\mathbb{F}|} + \sum_{i=1}^{d-1} \frac{1}{|\mathbb{F}|} \\
&\leq \frac{\lg(S)}{|\mathbb{F}|} + \sum_{i=0}^{d} \frac{2\lg(S)}{|\mathbb{F}|} + \sum_{i=1}^{d} \frac{1}{|\mathbb{F}|} \\
&= \frac{3d\lg(S) + d}{|\mathbb{F}|}
\end{aligned}$$

Efficiency

Efficiency: Communication $O(S_0 + d \cdot \lg(S))$

- Preprocessing: W', which has size $2^{s_0} = S_0$.
- Round i: $O(s_{i+1})$
  - One sumcheck: $O\left(\sum_{j=1}^{2s_{i+1}} \deg_j\left(f_i^{(j)}\right)\right) = O(s_{i+1})$

Efficiency: Communication $O(S_0 + d \cdot \lg(S))$

- Preprocessing: W', which has size $2^{s_0} = S_0$.
- Round i: $O(s_{i+1})$
  - One sumcheck: $O\left(\sum_{j=1}^{2s_{i+1}} \deg_j\left(f_i^{(j)}\right)\right) = O(s_{i+1})$

$$O\left(S_0 + \sum_{i=0}^{d-1} s_{i+1}\right) = O\left(S_0 + \sum_{i=0}^{d-1} \lg(S_i)\right) = O(S_0 + d \cdot \lg(S))$$

Efficiency: Verifier $O(S_0 + d \cdot \lg(S) + S + n)$

- Proportional to communication cost: $O(S_0 + d \cdot \lg(S))$
- Evaluating $\widetilde{\text{add}}$, $\widetilde{\text{mul}}$: $O(t)$ (bounded by $O(S)$)
- The input layer: $O(n)$

Efficiency: Verifier $O(S_0 + d \cdot \mathbf{lg}(S) + S + n)$

- Proportional to communication cost: $O(S_0 + d \cdot \lg(S))$
- Evaluating $\widetilde{\text{add}}, \widetilde{\text{mul}}$: $O(t)$ (bounded by $O(S)$)
- The input layer: $O(n)$

$$O(S_0 + d \cdot \lg(S) + t + n) = O(S_0 + d \cdot \lg(S) + S + n)$$

Efficiency: Prover Runtime per Sumcheck

$$\begin{aligned} O(2^{\ell} \cdot T) &= O(2^{2s_{i+1}} \cdot T) \\ &= O(S_{i+1}^2 \cdot (S_i + S_{i+1})) \\ &= O(S_{i+1}^3 + S_i S_{i+1}^2) \end{aligned}$$

Efficiency: Prover Runtime per Sumcheck

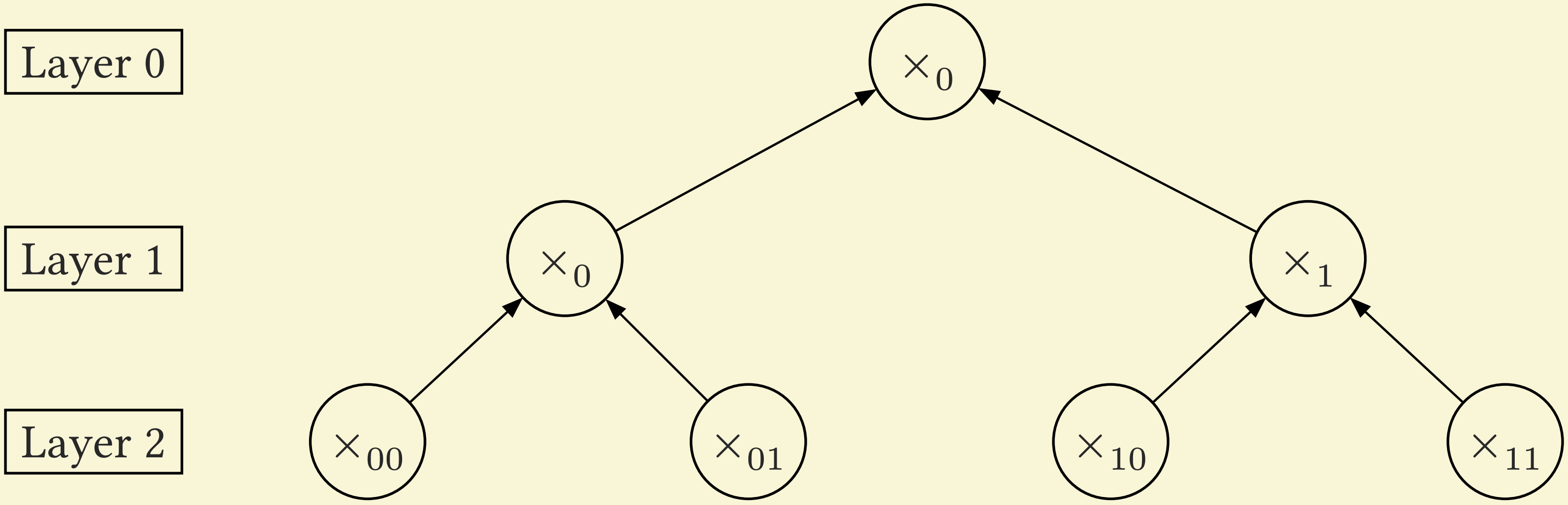
$$\begin{aligned} O(2^{\ell} \cdot T) &= O(2^{2s_{i+1}} \cdot T) \\ &= O(S_{i+1}^2 \cdot (S_i + S_{i+1})) \\ &= O(S_{i+1}^3 + S_i S_{i+1}^2) \end{aligned}$$

Efficiency: Prover Total

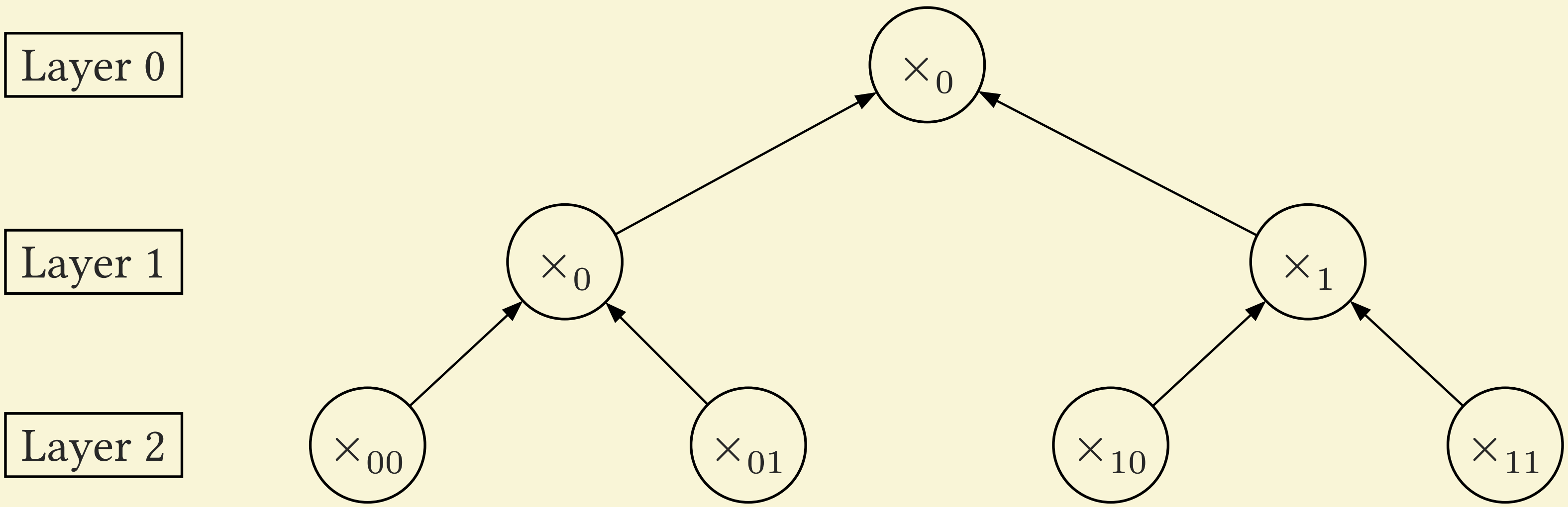
$$O\left(\sum_{i=0}^{d-1} (S_i S_{i+1}^2 + S_{i+1}^3)\right) = O(S^3)$$

Need for Speed - A GKR-Based Grand Product Proof

The Grand Product Proof



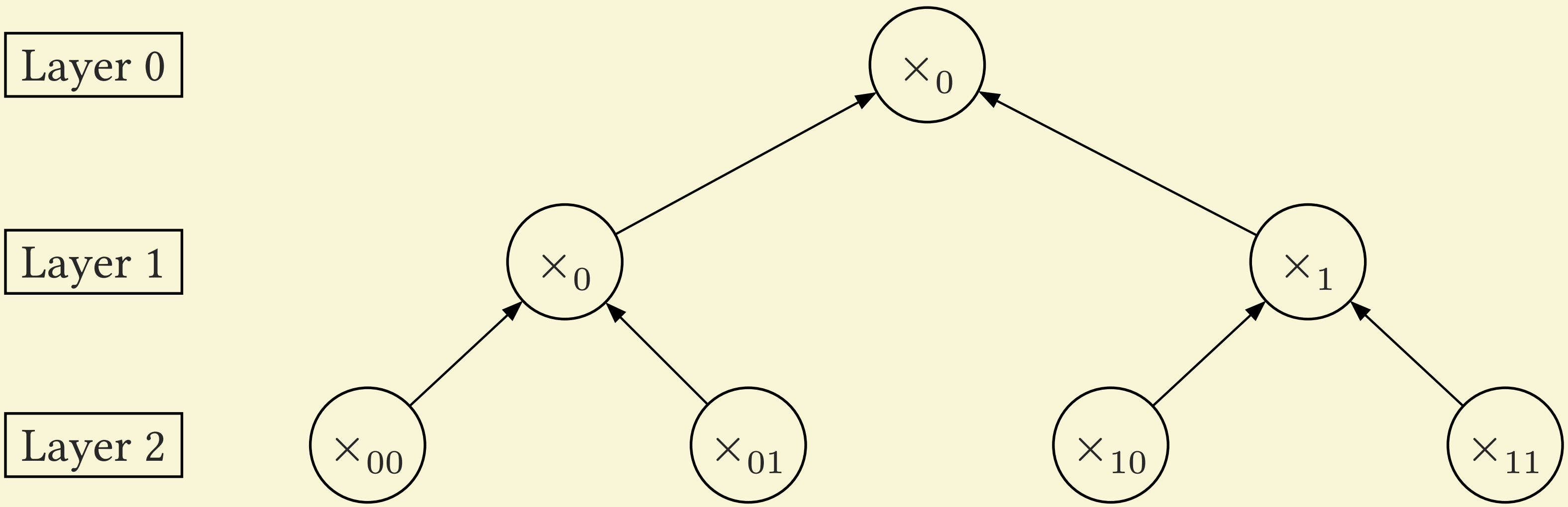
The Grand Product Proof



$$\widetilde{W}_i(\boldsymbol{a}) = \sum_{b \in \mathbb{B}^i} \tilde{\mathrm{eq}}(\boldsymbol{a}, \boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1)$$

$$s_i = i, \qquad S_i = 2^i$$

The Grand Product Proof



$$\widetilde{W}_i(\boldsymbol{a}) = \sum_{b \in \mathbb{B}^i} \tilde{\text{eq}}(\boldsymbol{a}, \boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1)$$

$$s_i = i, \qquad S_i = 2^i$$

$$S = \sum_{i=0}^{d} 2^i = 2^{d+1} - 1 = 2n - 1$$

The Sumcheck Polynomial

$$
\begin{aligned}
q(\boldsymbol{r}_{i-1}) &= \widetilde{W}_{i+1}(\boldsymbol{r}_{i-1} \parallel 0) + \alpha \widetilde{W}_{i+1}(\boldsymbol{r}_{i-1} \parallel 1) \\
&= \left(\sum_{\boldsymbol{b} \in \mathbb{B}^i} \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 0)}(\boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1) \right) + \\
&\quad \left(\sum_{\boldsymbol{b} \in \mathbb{B}^i} \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 1)}(\boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1) \right) \\
&= \sum_{\boldsymbol{b} \in \mathbb{B}^i} \Big(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 0)}(\boldsymbol{b}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 1)}(\boldsymbol{b}) \Big) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1)
\end{aligned}
$$

The Sumcheck Polynomial

$$
\begin{aligned}
q(\boldsymbol{r}_{i-1}) &= \widetilde{W}_{i+1}(\boldsymbol{r}_{i-1} \parallel 0) + \alpha \widetilde{W}_{i+1}(\boldsymbol{r}_{i-1} \parallel 1) \\
&= \left(\sum_{\boldsymbol{b} \in \mathbb{B}^i} \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 0)}(\boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1) \right) + \\
&\quad \left(\sum_{\boldsymbol{b} \in \mathbb{B}^i} \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 1)}(\boldsymbol{b}) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1) \right) \\
&= \sum_{\boldsymbol{b} \in \mathbb{B}^i} \left(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 0)}(\boldsymbol{b}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 1)}(\boldsymbol{b}) \right) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1)
\end{aligned}
$$

Sumcheck Polynomial:

$$
f_i(\boldsymbol{b}) = \left(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 0)}(\boldsymbol{b}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \parallel 1)}(\boldsymbol{b}) \right) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \parallel 1)
$$

Round Zero?

What about round zero? There's only a single gate.

Round Zero?

What about round zero? There's only a single gate.

We simply skip it!

Round Zero?

What about round zero? There's only a single gate.

We simply skip it!

$$\widetilde{W}_0(x) = \sum_{b \in \mathbb{B}^0} \tilde{\mathrm{eq}}(x, b) \cdot \widetilde{W}_1(b \parallel 0) \cdot \widetilde{W}_1(b \parallel 1)$$
$$= \tilde{\mathrm{eq}}(x, ()) \cdot \widetilde{W}_1(() \parallel 0) \cdot \widetilde{W}_1(() \parallel 1)$$
$$= \widetilde{W}_1(0) \cdot \widetilde{W}_1(1)$$

Full Protocol: Preprocessing

The prover sends evaluations $v_0^{(0)}, v_1^{(0)}$ claiming that $v_0^{(0)} = \widetilde{W}_1(0), v_1^{(0)} = \widetilde{W}_1(1)$ and thus $y = v_0^{(0)} \cdot v_1^{(0)}$.

$$\begin{array}{l} v_0^{(0)} := \widetilde{W}_1(0), \\ v_1^{(0)} := \widetilde{W}_1(1) \end{array} \xrightarrow{\qquad v_0^{(0)}, v_1^{(0)} \qquad} m_0 := v_0^{(0)} \cdot v_1^{(0)}$$

Full Protocol: Round i

$$f_i(\boldsymbol{b}) = \left(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{b}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{b})\right) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \,\|\, 0) \cdot \widetilde{W}_{i+1}(\boldsymbol{b} \,\|\, 1)$$

$$f_i(\boldsymbol{b}) \xleftarrow{\quad\quad\alpha\quad\quad} \alpha \in_R \mathbb{F}$$

$$\xleftrightarrow{\sum_{\boldsymbol{b},\boldsymbol{c} \in \mathbb{B}^i} f^i(\boldsymbol{b}) \stackrel{?}{=} m_i}$$

$$\begin{aligned} v_0^{(i)} &:= \widetilde{W}_{i+1}(\boldsymbol{r}_i \,\|\, 0), \\ v_1^{(i)} &:= \widetilde{W}_{i+1}(\boldsymbol{r}_i \,\|\, 1) \end{aligned} \xrightarrow{\quad v_0^{(i)}, v_1^{(i)} \quad} \begin{aligned} m_i \stackrel{?}{=} {} & \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{r}_i) \cdot v_0^{(i)} \cdot v_1^{(i)} + \\ & \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{r}_i) \cdot v_0^{(i)} \cdot v_1^{(i)} \end{aligned}$$

At the input layer d, $\mathcal{V}$ has two claims. $\mathcal{V}$ constructs $\widetilde{W}_d$ from $\boldsymbol{w}$ and perform a final check.

$$v_0^{(d-1)} \stackrel{?}{=} \widetilde{W}_d(r_{d-1} \parallel 0) \wedge$$
$$v_1^{(d-1)} \stackrel{?}{=} \widetilde{W}_d(r_{d-1} \parallel 1)$$

The GKR-Based Grand Product Proof - Efficiency

Linear Prover: Main idea

- If each round of sumcheck takes $O(2^{i-j})$ time, then:

$$O\left(\sum_{j=1}^{i} 2^{i-j}\right) = O\left(\sum_{j=1}^{i-1} 2^{i-j}\right) = O(2^i - 1) = O(2^i)$$

Linear Prover: Main idea

- If each round of sumcheck takes $O(2^{i-j})$ time, then:

$$O\left(\sum_{j=1}^{i} 2^{i-j}\right) = O\left(\sum_{j=1}^{i-1} 2^{i-j}\right) = O(2^i - 1) = O(2^i)$$

- Prover cost is dominated by the sumchecks, so this leads to:

$$O\left(\sum_{i=0}^{d} 2^i\right) = O(2^{d+1}) = O(2^d) = O(n)$$

Linear Prover: Use Lookup Tables

$$\begin{aligned}
\boldsymbol{a} &= \left(r_1, ..., r_j, t, x_1, ..., x_i\right), \\
f_i(\boldsymbol{a}) &= \left(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{a}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{a})\right) \cdot \widetilde{W}_{i+1}(\boldsymbol{a}) \cdot \widetilde{W}_{i+1}(\boldsymbol{a}) \\
&= \left(\hat{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{a}) + \alpha \cdot \hat{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{a})\right) \cdot \widehat{W}_{i+1}(\boldsymbol{a}) \cdot \widehat{W}_{i+1}(\boldsymbol{a})
\end{aligned}$$

Linear Prover: Use Lookup Tables

$$\boldsymbol{a} = (r_1, ..., r_j, t, x_1, ..., x_i),$$

$$\begin{aligned} f_i(\boldsymbol{a}) &= \left(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{a}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{a})\right) \cdot \widetilde{W}_{i+1}(\boldsymbol{a}) \cdot \widetilde{W}_{i+1}(\boldsymbol{a}) \\ &= \left(\hat{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{a}) + \alpha \cdot \hat{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{a})\right) \cdot \widehat{W}_{i+1}(\boldsymbol{a}) \cdot \widehat{W}_{i+1}(\boldsymbol{a}) \end{aligned}$$

- Constant time evaluation of f_i!
- $\deg(f_i) + 1 = 4$ points, still constant time.

Linear Prover: Use Lookup Tables

$$\begin{aligned}
\boldsymbol{a} &= (r_1, ..., r_j, t, x_1, ..., x_i), \\
f_i(\boldsymbol{a}) &= \left(\tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{a}) + \alpha \cdot \tilde{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{a})\right) \cdot \widetilde{W}_{i+1}(\boldsymbol{a}) \cdot \widetilde{W}_{i+1}(\boldsymbol{a}) \\
&= \left(\hat{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 0)}(\boldsymbol{a}) + \alpha \cdot \hat{\mathrm{eq}}_{(\boldsymbol{r}_{i-1} \,\|\, 1)}(\boldsymbol{a})\right) \cdot \widehat{W}_{i+1}(\boldsymbol{a}) \cdot \widehat{W}_{i+1}(\boldsymbol{a})
\end{aligned}$$

- Constant time evaluation of f_i!
- $\deg(f_i) + 1 = 4$ points, still constant time.

Assuming $\hat{\mathrm{eq}}$, $\widehat{W}_{i+1}$ can be computed in $O(2^{i-j})$ time...

Linear Prover: $\hat{\mathrm{eq}}_j$

$$\boldsymbol{v} = (\boldsymbol{r}_{i-1} \parallel 0) \vee (\boldsymbol{r}_{i-1} \parallel 1),$$

$$\hat{\mathrm{eq}}_j\left[\left(b_{j+1}, ..., b_\ell\right)\right] = v_j^{-1} \cdot \hat{\mathrm{eq}}_{j-1}\left[\left(1, b_{j+1}, ..., b_\ell\right)\right] \cdot v_j r_j + \left(1 - v_j\right)\left(1 - r_j\right)$$

- $\hat{\mathrm{eq}}_0$ can be computed in $O\left(2^\ell\right) = O\left(2^i\right)$ time
- $\hat{\mathrm{eq}}_j$ can be computed in $O\left(2^{\ell-j}\right) = O\left(2^{i-j}\right)$ time.

Computing $\hat{\mathrm{W}}_j$:

$$\begin{aligned}\widehat{W}_0\left[\boldsymbol{x} \in \mathbb{B}^\ell\right] &:= W_{i+1}\\ \widehat{W}_j\left[\left(x_{j+1}, \ldots, x_\ell\right)\right] &:= \tilde{\mathrm{eq}}_0\left(r_j\right) \cdot \widehat{W}_{j-1}\left[\left(0, x_{j+1}, \ldots, x_\ell\right)\right] +\\ &\quad\ \tilde{\mathrm{eq}}_1\left(r_j\right) \cdot \widehat{W}_{j-1}\left[\left(1, x_{j+1}, \ldots, x_\ell\right)\right]\end{aligned}$$

Computing $\hat{\mathrm{W}}_j$**:**

$$\widehat{W}_0\left[\boldsymbol{x} \in \mathbb{B}^\ell\right] := W_{i+1}$$

$$\widehat{W}_j\left[\left(x_{j+1}, ..., x_\ell\right)\right] := \tilde{\mathrm{eq}}_0\left(r_j\right) \cdot \widehat{W}_{j-1}\left[\left(0, x_{j+1}, ..., x_\ell\right)\right] + \tilde{\mathrm{eq}}_1\left(r_j\right) \cdot \widehat{W}_{j-1}\left[\left(1, x_{j+1}, ..., x_\ell\right)\right]$$

$$O\left(2^{\ell-j}\right) = O\left(2^{i+1-j}\right) = O\left(2^{i-j}\right) \text{ time!}$$

Computing $\hat{\mathrm{W}}_j$:

$$\widehat{W}_0[\boldsymbol{x} \in \mathbb{B}^\ell] := W_{i+1}$$

$$\begin{aligned}\widehat{W}_j[(x_{j+1}, ..., x_\ell)] := \tilde{\mathrm{eq}}_0(r_j) \cdot \widehat{W}_{j-1}[(0, x_{j+1}, ..., x_\ell)] +\\ \tilde{\mathrm{eq}}_1(r_j) \cdot \widehat{W}_{j-1}[(1, x_{j+1}, ..., x_\ell)]\end{aligned}$$

$$O(2^{\ell-j}) = O(2^{i+1-j}) = O(2^{i-j}) \text{ time!}$$

Using $\hat{\mathrm{W}}_j$:

$$\begin{aligned}\widetilde{W}(r_1, ..., r_{j-1}, t, x_{j+1}, ..., x_\ell) = \tilde{\mathrm{eq}}_0(t) \cdot \widehat{W}_{j-1}[(0, x_{j+1}, ..., x_\ell)] +\\ \tilde{\mathrm{eq}}_1(t) \cdot \widehat{W}_{j-1}[(1, x_{j+1}, ..., x_\ell)]\end{aligned}$$

Linear Prover!

Communication $O(\lg^2(n))$

- Round i: $O(s_{i+1})$
  - One sumcheck: $O\left(\sum_{j=1}^{2s_{i+1}} \deg_j(f_i)\right) = O(i)$

$$O\left(\sum_{i=1}^{d-1} i\right) = O\left(\frac{d(d+1)}{2}\right) = O(d^2) = O(\lg^2(n))$$

Verifier $O(\lg^2(n) + n)$

$$\begin{aligned}
O(\lg^2(n) + t + n) &= O\left(\lg^2(n) + \sum_{i=1}^{d-1} i + n\right) \\
&= O(\lg^2(n) + \lg^2(n) + n) \\
&= O(\lg^2(n) + n)
\end{aligned}$$

Conclusion

- *Linear* prover
- Almost *sublinear* verifier
- Sublinear communication costs
- All in the size of the *witness*, not just the circuit

Conclusion

- *Linear* prover
- Almost *sublinear* verifier
- Sublinear communication costs
- All in the size of the *witness*, not just the circuit

Excellent construction for SNARKs (Spartan, Lasso)

Fin