Cached quotients and lookups

Ariel Gabizon

31. januar 2023

Constraints vs Lookups

Example: Check $\mathbf{0 \le x \le 2^n - 1}$

Constraints vs Lookups

Example: Check $0 \leq x \leq 2^n - 1$

Constraint approach:
Prover sends $b_0, \ldots, b_{n-1}$. Shows:

- $\forall i, b_i \in \{0, 1\}$
- $\sum_i b_i 2^i = x$.

Constraints vs Lookups

Example: Check $\mathbf{0 \leq x \leq 2^n - 1}$

Constraint approach:
Prover sends $\mathbf{b_0, \ldots, b_{n-1}}$. Shows:

- $\forall \mathbf{i}, \mathbf{b_i} \in \{\mathbf{0}, \mathbf{1}\}$
- $\sum_{\mathbf{i}} \mathbf{b_i 2^i} = \mathbf{x}$.

Requires $\mathbf{n + 1}$ constraints.

Lookup approach

Preprocess table $\mathbf{T} = \{\mathbf{0}, \ldots, \mathbf{2}^{n} - \mathbf{1}\}$.Let $\mathbf{N} := |\mathbf{T}|$.
Devise protocol to check $x \in \mathbf{T}$.

Lookup approach

Preprocess table $\mathbf{T} = \{\mathbf{0}, \ldots, \mathbf{2^n - 1}\}$.Let $\mathbf{N} := |\mathbf{T}|$.
Devise protocol to check $x \in \mathbf{T}$.

Old results - good when amortized:
Thm[plookup]: Can check m different x's are in $\mathbf{T}$ in $\mathbf{O}(m + \mathbf{N})$ constraints.

Lookup approach

Preprocess table $\mathbf{T} = \{\mathbf{0}, \ldots, \mathbf{2^n - 1}\}$.Let $\mathbf{N} := |\mathbf{T}|$.
Devise protocol to check $\mathbf{x} \in \mathbf{T}$.

Old results - good when amortized:
Thm[plookup]: Can check $\mathbf{m}$ different $\mathbf{x}$'s are in $\mathbf{T}$ in $\mathbf{O}(\mathbf{m} + \mathbf{N})$ constraints.

New results - prover doesn't pay for table size!!
Thm [Caulk...$\mathfrak{cq}$]: After $\mathbf{O}(\mathbf{N} \log \mathbf{N})$ preprocessing, can check $\mathbf{x} \in \mathbf{T}$, in $\mathbf{O}(\mathbf{1})$ constraints.

Rest of talk: explain main technical component of new works - *cached quotients*

Rest of talk: explain main technical component of new works - *cached quotients*

First - a brief recap of polynomial commitment schemes..

The KZG Polynomial commitment scheme

$\mathbf{G}$ - generator of pairing friendly elliptic curve group.

$\mathbf{srs := 1 \cdot G, x \cdot G, \ldots, x^d \cdot G}$, for random $\mathbf{x} \in \mathbb{F}$.

The KZG Polynomial commitment scheme

$\mathbf{G}$ - generator of pairing friendly elliptic curve group.

$\mathbf{srs} := \mathbf{1} \cdot \mathbf{G}, x \cdot \mathbf{G}, \ldots, x^{\mathbf{d}} \cdot \mathbf{G}$, for random $x \in \mathbb{F}$.

For $\mathbf{f} \in \mathbb{F}[\mathbf{X}]$ of degree $\mathbf{d}$:

$$\mathbf{cm}(\mathbf{f}) := \mathbf{f}(x) \cdot \mathbf{G}$$

The KZG Polynomial commitment scheme

$\mathbf{G}$ - generator of pairing friendly elliptic curve group.

$\mathbf{srs} := \mathbf{1} \cdot \mathbf{G}, x \cdot \mathbf{G}, \ldots, x^{d} \cdot \mathbf{G}$, for random $x \in \mathbb{F}$.

For $f \in \mathbb{F}[X]$ of degree d:

$$\mathbf{cm}(f) := f(x) \cdot \mathbf{G}$$

Central Feature: Given $\mathbf{cm}(f)$ and any $\alpha \in \mathbb{F}$;
there is short proof for correctness of $z = f(\alpha)$.

The KZG Polynomial commitment scheme

$\textbf{srs} := 1 \cdot G, x \cdot G, \ldots, x^d \cdot G$, for random $x \in \mathbb{F}$.
$\textbf{cm}(f) := f(x) \cdot G$

Nice features:

The KZG Polynomial commitment scheme

$\mathbf{srs} := \mathbf{1} \cdot \mathbf{G}, \boldsymbol{x} \cdot \mathbf{G}, \ldots, \boldsymbol{x}^{\mathbf{d}} \cdot \mathbf{G}$, for random $\boldsymbol{x} \in \mathbb{F}$.
$\mathbf{cm}(\mathbf{f}) := \mathbf{f}(\boldsymbol{x}) \cdot \mathbf{G}$

Nice features:

- **Linearity:** $\mathbf{cm}(\mathbf{f} + \mathbf{g}) = \mathbf{cm}(\mathbf{f}) + \mathbf{cm}(\mathbf{g})$

The KZG Polynomial commitment scheme

$\mathbf{srs} := \mathbf{1} \cdot \mathbf{G}, x \cdot \mathbf{G}, \ldots, x^{d} \cdot \mathbf{G}$, for random $x \in \mathbb{F}$.
$\mathbf{cm}(f) := f(x) \cdot \mathbf{G}$

Nice features:

- **Linearity:** $\mathbf{cm}(f+g) = \mathbf{cm}(f) + \mathbf{cm}(g)$
- **Product checks:** Given $\mathbf{cm}(f_1), \mathbf{cm}(f_2), \mathbf{cm}(g_1), \mathbf{cm}(g_2)$ can check $f_1(X) f_2(X) \stackrel{?}{\equiv} g_1(X) g_2(X)$ via pairings. (Secure in the Algebraic Group Model)

Cached Quotients - a motivating example from **Caulk**[ZBKMN]

Cached Quotients - a motivating example from **Caulk**[ZBKMN]

$Z_T(X) = \prod_{a \in T}(X - a)$ a vanishing polynomial of a subset $T \subset \mathbb{F}$.

Cached Quotients - a motivating example from **Caulk**[ZBKMN]

$Z_T(X) = \prod_{a \in T}(X - a)$ a vanishing polynomial of a subset $T \subset \mathbb{F}$.

$\mathbf{cm}(Z_T), \mathbf{cm}(f)$ given to verifier.

Cached Quotients - a motivating example from **Caulk**[ZBKMN]

$Z_T(X) = \prod_{a \in T}(X - a)$ a vanishing polynomial of a subset $T \subset \mathbb{F}$.

$\mathbf{cm}(Z_T), \mathbf{cm}(f)$ given to verifier.
Prover wants to show $f = Z_S$ for some $S \subset T$.

Cached Quotients - a motivating example from **Caulk**[ZBKMN]

$Z_T(X) = \prod_{a \in T}(X - a)$ a vanishing polynomial of a subset $T \subset \mathbb{F}$.

$\mathbf{cm}(Z_T), \mathbf{cm}(f)$ given to verifier.
Prover wants to show $f = Z_S$ for some $S \subset T$.

Can we do this in $O(|S|)$ prover operations?(think $|S| \ll |T|$)

Cached quotients idea:

The quotient $Z_{T\setminus S}(X) = \frac{Z_T(X)}{Z_S(X)}$ is a "witness" to $S \subset T$.

Cached quotients idea:

The quotient $Z_{T \setminus S}(X) = \frac{Z_T(X)}{Z_S(X)}$ is a "witness" to $S \subset T$.

- Enough to compute **commitment** to $Z_{T \setminus S}$.

Cached quotients idea:

The quotient $Z_{T\setminus S}(X) = \frac{Z_T(X)}{Z_S(X)}$ is a "witness" to $S \subset T$.

- Enough to compute **commitment** to $Z_{T\setminus S}$.
- This commitment is a **sparse combination** of commitments we can **precompute**.

details in next slide..

For each $i \in T$, let $g_i(X) := Z_{T \setminus \{i\}}(X)$.

For each $i \in T$, let $g_i(X) := Z_{T \setminus \{i\}}(X)$.

We have [Tomescu et. al]

$$Z_{T \setminus S}(X) = \sum_{i \in S} c_i \cdot g_i(X)$$

for some $c_i \in \mathbb{F}$.

For each $\mathbf{i} \in \mathbf{T}$, let $\mathbf{g_i(X) := Z_{T \setminus \{i\}}(X)}$.

We have [Tomescu et. al]

$$Z_{T \setminus S}(X) = \sum_{i \in S} c_i \cdot g_i(X)$$

for some $\mathbf{c_i} \in \mathbb{F}$.

We precompute $\mathbf{cm(Z_T), \{cm(g_i)\}_{i \in T}}$.

Prover then computes in $|S|$ operations:

$$\pi := \mathbf{cm}(Z_{T \setminus S}) = \sum_{i \in S} c_i \cdot \mathbf{cm}(g_i)$$

Prover then computes in $|S|$ operations:

$$\pi := \mathbf{cm}(Z_{T \setminus S}) = \sum_{i \in S} c_i \cdot \mathbf{cm}(g_i)$$

Verifier checks with pairing that:

$$e(\mathbf{cm}(f), \pi) = e(\mathbf{cm}(Z_T), 1 \cdot G)$$