Speeding up SNARKs with cached quotients

Ariel Gabizon

4. april 2023

The triology of pairing-based SNARKs

The triology of pairing-based SNARKs

1. **A new hope (for SNARKs, not the universe)** - [Groth10,**GGPR**,...,Groth16]

The triology of pairing-based SNARKs

1. **A new hope (for SNARKs, not the universe)** - [Groth10,**GGPR**,...,Groth16]

2. **The polynomial commitment scheme strikes back** - [vsql,**Sonic**,Plonk,Marlin,...]

The triology of pairing-based SNARKs

1. **A new hope (for SNARKs, not the universe)** - [Groth10,**GGPR**,...,Groth16]
2. **The polynomial commitment scheme strikes back** - [vsql,**Sonic**,Plonk,Marlin,...]
3. **Return of the pairing** - [Caulk,...]

First a short KZG Reminder..

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

Nice features:

First a short KZG Reminder..

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

Nice features:

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

First a short KZG Reminder..

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

Nice features:

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

The first scene of chapter three from

Caulk

The first scene of chapter three from

Caulk

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

The first scene of chapter three from **Caulk**

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

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

The first scene of chapter three from **Caulk**

$Z_T(X) = \prod_{a \in T}(X - a)$ 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 $S \subset T$.

The first scene of chapter three from **Caulk**

$Z_T(X) = \prod_{a \in T}(X - a)$ 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 $S \subset T$.

"Do it in $O(|S|)$ *prover operations or be thrown in the pit!"* (think $|S| \ll |T|$)

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

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 $Z_{T \setminus S}$.

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]

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

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:

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

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:

$$e(\mathbf{cm}(f), \pi) = e(\mathbf{cm}(Z_T), [1])$$

Sparse polynomials

parameters $n \ll N$.

Sparse polynomials

parameters $\mathbf{n} \ll \mathbf{N}$.

$\mathbf{B} = \{\mathbf{B_1(X)}, \ldots, \mathbf{B_N(X)}\}$ linearly independent set of polynomials.

Sparse polynomials

parameters $n \ll N$.

$B = \{B_1(X), \ldots, B_N(X)\}$ linearly independent set of polynomials.

Dfn: $A \in \mathbb{F}[X]$ is n-*sparse* in base B, if we can write

$$A(X) = \sum_{i \in [N]} a_i \cdot B_i(X)$$

where only n a_i's are non-zero.

Sparse polynomials

parameters $n \ll N$.

$B = \{B_1(X), \ldots, B_N(X)\}$ linearly independent set of polynomials.

Dfn: $A \in \mathbb{F}[X]$ is n-*sparse* in base B, if we can write

$$A(X) = \sum_{i \in [N]} a_i \cdot B_i(X)$$

where only n a_i's are non-zero.

Default case: B is Lagrange base of subgroup of size N.

Committing to sparse polynomials

We can precompute the KZG commitments for $\mathbf{B}$:
$\mathbf{srs_B} := \{\mathbf{cm}(\mathbf{B_1}), \ldots, \mathbf{cm}(\mathbf{B_N})\}$

Committing to sparse polynomials

We can precompute the KZG commitments for B:
$\mathbf{srs}_B := \{\mathbf{cm}(B_1), \ldots, \mathbf{cm}(B_N)\}$

Later, for n-sparse $A(X)$ we can compute

$$\mathbf{cm}(A) = \sum_{i \in [N], a_i \neq 0} a_i \cdot \mathbf{cm}(B_i)$$

in n operations.

Cached quotients method

Scenario: $T(X), Z(X)$ preprocessed polys.
$\mathbf{deg}(Z) = N$.

Cached quotients method

Scenario: $T(X), Z(X)$ preprocessed polys.
$\mathbf{deg}(Z) = N$.

Input: n-sparse $A(X)$, and some $R(X)$ of deg$< N$.
V has $\mathbf{cm}(A), \mathbf{cm}(R)$.

Cached quotients method

Scenario: $T(X), Z(X)$ preprocessed polys.
$\mathbf{deg}(Z) = N$.

Input: n-sparse $A(X)$, and some $R(X)$ of deg$< N$.
V has $\mathbf{cm}(A), \mathbf{cm}(R)$.

Want to prove to V that:

$$A(X)T(X) \equiv R(X) \quad \mathbf{mod} \;\; Z(X)$$

using $O(n)$ *prover operations*.

Cached quotients method

There exists quotient $\mathbf{Q}(\mathbf{X})$ such that
$\mathbf{A} \cdot \mathbf{T} = \mathbf{Z} \cdot \mathbf{Q} + \mathbf{R}$.

Cached quotients method

There exists quotient $Q(X)$ such that
$A \cdot T = Z \cdot Q + R$.

We'll compute $\mathbf{cm}(Q)$ in n operations:

Cached quotients method

There exists quotient $Q(X)$ such that $A \cdot T = Z \cdot Q + R$.

We'll compute $\mathbf{cm}(Q)$ in n operations:

preprocessing: For each $i \in [N]$, compute $\mathbf{cm}(Q_i)$ such that for some $R_i(X) \in \mathbb{F}_{<N}[X]$

$$B_i(X) \cdot T(X) = Q_i(X) \cdot Z(X) + R_i(X)$$

Cached quotients method

There exists quotient $Q(X)$ such that
$A \cdot T = Z \cdot Q + R$.

We'll compute $\mathbf{cm}(Q)$ in n operations:

preprocessing: For each $i \in [N]$, compute
$\mathbf{cm}(Q_i)$ such that for some $R_i(X) \in \mathbb{F}_{<N}[X]$

$$B_i(X) \cdot T(X) = Q_i(X) \cdot Z(X) + R_i(X)$$

Also precompute $\mathbf{cm}(Z), \mathbf{cm}(T)$

$\mathbf{Q}$ "inherits" $\boldsymbol{A}$'s sparseness

$\boldsymbol{A}(\mathbf{X}) = \sum_i \mathbf{a_i B_i}(\mathbf{X})$
After preprocessing, prover can compute and send

$$\mathbf{cm}(\mathbf{Q}) = \sum_{i \in [N], a_i \neq 0} \mathbf{a_i} \cdot \mathbf{cm}(\mathbf{Q_i})$$

$\mathbf{Q}$ "inherits" $\boldsymbol{A}$'s sparseness

$A(X) = \sum_i a_i B_i(X)$
After preprocessing, prover can compute and send

$$\mathbf{cm}(Q) = \sum_{i \in [N], a_i \neq 0} a_i \cdot \mathbf{cm}(Q_i)$$

Verifier can then check with pairings:

$$AT \stackrel{?}{=} QZ + R$$

Application 1: lookups

Preprocessed table T of size N, witness f of size n, $n \ll N$. Want to check $f_i \in T$ for each $i \in [n]$

Application 1: lookups

Preprocessed table T of size N, witness f of size n, $n \ll N$. Want to check $f_i \in T$ for each $i \in [n]$

[Caulk,...,$\mathfrak{cq}$]: Can be done in prover time $O(n \log n)$.
(improved plookup's $O(N \cdot \log N)$)

Application 1: lookups

Preprocessed table $\mathbf{T}$ of size $\mathbf{N}$, witness $\mathbf{f}$ of size $\mathfrak{n}$, $\mathfrak{n} \ll \mathbf{N}$. Want to check $\mathbf{f_i} \in \mathbf{T}$ for each $\mathbf{i} \in [\mathfrak{n}]$

[Caulk,...,$\mathfrak{cq}$]: Can be done in prover time $\mathbf{O}(\mathfrak{n} \log \mathfrak{n})$.
(improved plookup's $\mathbf{O}(\mathbf{N} \cdot \log \mathbf{N})$)

proof sketch: In log-derivative lookup [Eagen,Haböck,..] prover multiplies $\mathfrak{n}$-sparse poly with a preprocessed poly representing the table - *good fit for cached quotients method.*

Application 2: lincheck

Fixed $n \times n$ matrix M.

Application 2: lincheck

Fixed $n \times n$ matrix M.

Prover has poly $f \in \mathbb{F}_{<n}[X]$. Verifier $\mathbf{cm}(f)$.
$a := f|_H$ for subgroup H of size n.

Application 2: lincheck

Fixed $\mathbf{n} \times \mathbf{n}$ matrix $\mathbf{M}$.

Prover has poly $\mathbf{f} \in \mathbb{F}_{<\mathbf{n}}[\mathbf{X}]$. Verifier $\mathbf{cm}(\mathbf{f})$.
$\mathbf{a} := \mathbf{f}|_{\mathbf{H}}$ for subgroup $\mathbf{H}$ of size $\mathbf{n}$.

Prover wants to show $\mathbf{M} \cdot \mathbf{a} = \mathbf{0}$ *in* $\mathbf{O}(\mathbf{n})$ *operations.*

Application 2: lincheck

Fixed $n \times n$ matrix M.

Prover has poly $f \in \mathbb{F}_{<n}[X]$. Verifier $\mathbf{cm}(f)$.
$a := f|_H$ for subgroup H of size n.

Prover wants to show $M \cdot a = \mathbf{0}$ *in* $O(n)$ *operations.*

Let $L_1, \ldots, L_n$ be a Lagrange basis for H.

Reducing to cached quotients:

Represent M by degree $\sim n^2$ polynomial

$$M(X) := \sum_{i,j \in [n]} M_{i,j} L_i(X^n) L_j(X)$$

Let $Z(X) := X^{n^2} - 1$.

Reducing to cached quotients:

Represent M by degree $\sim n^2$ polynomial

$$M(X) := \sum_{i,j \in [n]} M_{i,j} L_i(X^n) L_j(X)$$

Let $Z(X) := X^{n^2} - 1$.

Let $A(X) := f(X^n)$, $R := A \cdot M \mod Z$.

Reducing to cached quotients:

Represent M by degree $\sim n^2$ polynomial

$$M(X) := \sum_{i,j \in [n]} M_{i,j} L_i(X^n) L_j(X)$$

Let $Z(X) := X^{n^2} - 1$.

Let $A(X) := f(X^n)$, $R := A \cdot M \mod Z$.

We have

$$R(X) = \sum_{j \in [n]} L_j(X) \sum_{i \in [n]} a_i \cdot M_{i,j} L_i(X^n).$$

Reducing to cached quotients:

Represent M by degree $\sim n^2$ polynomial

$$M(X) := \sum_{i,j\in[n]} M_{i,j} L_i(X^n) L_j(X)$$

Let $Z(X) := X^{n^2} - 1$.

Let $A(X) := f(X^n)$, $R := A \cdot M \mod Z$.

We have

$$R(X) = \sum_{j\in[n]} L_j(X) \sum_{i\in[n]} a_i \cdot M_{i,j} L_i(X^n).$$

So $M \cdot a = 0$ iff $R(X) \equiv 0 \mod X^n$.

Note that A is n-sparse in the basis $\{L_i(X^n)\}$

Note that A is n-sparse in the basis $\{L_i(X^n)\}$

Use cached quotients twice to show

1. $A(X) \cdot M(X) \equiv R(X) \mod Z(X)$.
2. $R(X) \equiv 0 \mod X^n$.

Note that A is n-sparse in the basis $\{L_i(X^n)\}$

Use cached quotients twice to show

1. $A(X) \cdot M(X) \equiv R(X) \mod Z(X)$.
2. $R(X) \equiv 0 \mod X^n$.

Important point: $R(X)$ is sparse in appropriate base of remainders in multiplication of M modulu Z.

Generalizing: The "cached commitments methodology"

Given a polynomial IOP using prover poly $\mathbf{f}$ that

- has high degree, but
- *low sparsity* in a preknown basis of polynomials $\mathbf{B_1, \ldots, B_d}$
- Is only used in degree$\leq \mathbf{2}$ verifier equations.

we can

Generalizing: The "cached commitments methodology"

Given a polynomial IOP using prover poly $\mathbf{f}$ that

- has high degree, but
- *low sparsity* in a preknown basis of polynomials $\mathbf{B_1, \ldots, B_d}$
- Is only used in degree $\leq \mathbf{2}$ verifier equations.

we can

1. Precompure the KZG commitments to $\mathbf{B_1, \ldots, B_d}$.
2. In protocol time, only compute *commitment* to $\mathbf{f}$ from pre-computed commitments
3. Use pairings to directly check verifier equations between commitments.