An efficient polynomial commitment scheme for multiple points and polynomials

Justin Drake
Ethereum Foundation

Ariel Gabizon
AZTEC Protocol

Zachary J. Williamson
AZTEC Protocol

January 23, 2020

Abstract

We present an enhanced version of the Kate, Zaverucha and Goldberg polynomial commitment scheme [KZG10] where a single group element can be an opening proof for multiple polynomials each evaluated at a different arbitrary subset of points.

As a sample application we "plug in" this scheme into the PLONK proving system[GWC19] to obtain improved proof size and prover run time at the expense of additional verifier $\mathbb{G}_2$ operations and pairings, and additional $\mathbb{G}_2$ SRS elements.

1 Introduction

Polynomial commitment schemes (PCS) are an important primitive in arguably all[1] practical succinct proving systems (see [CHM+19, GWC19, BFS19] for formalizations of their role). They can "force" a prover to answer verifier queries according to a fixed polynomial of bounded degree.

The more straightforward version of this primitive contains an initial prover message $\mathsf{com}(f)$ corresponding to the commitment to a polynomial f. Then, whenever the prover sends a value $s \in \mathbb{F}$ which is allegedly the value $f(z)$ for z known to the verifier, the prover will send a corresponding opening proof π that this is indeed the case. In a protocol where this primitive is used for several polynomials and several evaluation points, prover run time and proof length will increase with each of these opening proofs.

1.1 Previous work and our results

We take the PCS of Kate, Zaverucha and Goldberg [KZG10] as our starting point. Their scheme is pairing based and an opening proof π consists of a single $\mathbb{G}_1$ group element. We ask whether we can get a variant of this scheme where a single opening proof (in the sense of one $\mathbb{G}_1$ element) can prove correctness of multiple polynomial evaluations.

[1] While QAP based proving systems like [PHGR16, Gro16] are not traditionally formulated as using a PCS, they can be interpreted as using a PCS that also forces the committed polynomial to be in a certain subspace. This is related to the linear interactive proof framework of [BCI+13].

Several works [MBKM19, Gab19, CHM+19, GWC19] (starting from [MBKM19]) have noticed that it possible to modify the PCS of [KZG10] in the random oracle model to allow for one opening proof for several polynomials *at the same point* $z \in \mathbb{F}$. [KZG10] give in their paper a less known version of their scheme allowing for one opening proof for one polynomial at several evaluation points.

In this paper, we give a scheme where a single group element can be the opening proof for multiple evaluation points and polynomials. We compare the performance of our PCS compared to a more straightforward batched version of the [KZG10] scheme as in [GWC19]. For simplicity, we look at the restricted case where we want to open t polynomials all with the same degree bound n, each at one *distinct* point. See Lemma 3.3 for the more detailed efficiency properties in the general case (where each polynomial is opened at a subset of points, and the subsets may repeat).

Table 1: Comparison of opening complexity for t polynomials on t distinct points. In prover/verifier work columns $\mathbb{G}_i$ means scalar multiplication in $\mathbb{G}_i$, $\mathbb{F}$ means addition or multiplication in $\mathbb{F}$, and $\mathbf{P}$ means pairing.

	SRS size	**prover work**	**proof length**	**verifier work**
KZG as in [GWC19]	n $\mathbb{G}_1$, 2 $\mathbb{G}_2$	$t \cdot n$ $\mathbb{G}_1$, $O(t \cdot n \log n)$ $\mathbb{F}$	t $\mathbb{G}_1$	$3t-2$ $\mathbb{G}_1$, 2 $\mathbf{P}$
This work	n $\mathbb{G}_1$, $t+1$ $\mathbb{G}_2$	n $\mathbb{G}_1$, $O(t \cdot n + n \log n)$ $\mathbb{F}$	1 $\mathbb{G}_1$	$t-1$ $\mathbb{G}_1$, $2t$ $\mathbb{G}_2$, $t+1$ $\mathbf{P}$

Application to PLONK: The PLONK proving system [GWC19] allows generating proofs of knowledge for assignments to fan-in two arithmetic circuits with a universal and updatable SRS (see the paragraph on this topic in Section 2.1). Most of the prover computation involves committing to several polynomials and opening them at two distinct evaluation points. Plugging in our PCS to PLONK allows saving in proof length and prover work related to the opening proof of the second evaluation point (we do not give full details, but all that is needed is repreating the transformation of Lemma 4.7 in [GWC19] using the PCS of Lemma 3.3 here instead of the PCS used there to obtain the new result).

We compare the PLONK scheme when using the [KZG10]-based PCS in [**?**] and the PCS of this paper in Table 2. As in [GWC19] we present look at two versions of PLONK where one optimizes fast proving, and the other small proof length.

Table 2: Comparison of PLONK efficiency for fan-in two circuit with n gates.

	SRS size	**prover group exponentations**	**proof length**	verifier work
[GWC19] (fast)	n $\mathbb{G}_1$, 2 $\mathbb{G}_2$	$9n$ $\mathbb{G}_1$ exp	9 $\mathbb{G}_1$, 7 $\mathbb{F}$	18 $\mathbb{G}_1$, 2 $\mathbf{P}$
[GWC19] (small)	$3n$ $\mathbb{G}_1$, 2 $\mathbb{G}_2$	$11n$ $\mathbb{G}_1$ exp	7 $\mathbb{G}_1$, 7 $\mathbb{F}$	16 $\mathbb{G}_1$, 2 $\mathbf{P}$
This work (fast)	n $\mathbb{G}_1$, 3 $\mathbb{G}_2$	$8n$ $\mathbb{G}_1$ exp	8 $\mathbb{G}_1$ 7 $\mathbb{F}$	18 $\mathbb{G}_1$, 4 $\mathbb{G}_2$, 3 $\mathbf{P}$
This work (small)	$3n$ $\mathbb{G}_1$, 3 $\mathbb{G}_2$	$10n$ $\mathbb{G}_1$ exp	6 $\mathbb{G}_1$ 7 $\mathbb{F}$	16 $\mathbb{G}_1$, 4 $\mathbb{G}_2$, 3 $\mathbf{P}$

2 Preliminaries

2.1 Terminology and Conventions

We assume our field $\mathbb{F}$ is of prime order. We denote by $\mathbb{F}_{<d}[X]$ the set of univariate polynomials over $\mathbb{F}$ of degree smaller than d. We assume all algorithms described receive as an implicit parameter the security parameter λ.

Whenever we use the term "efficient", we mean an algorithm running in time $\mathsf{poly}(\lambda)$. Furthermore, we assume an "object generator" $\mathcal{O}$ that is run with input λ before all protocols, and returns all fields and groups used. Specifically, in our protocol $\mathcal{O}(\lambda) = (\mathbb{F}, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_t, e, g_1, g_2, g_t)$ where

- $\mathbb{F}$ is a prime field of super-polynomial size $r = \lambda^{\omega(1)}$.
- $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_t$ are all groups of size r, and e is an efficiently computable non-degenerate pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_t$.
- g_1, g_2 are uniformly chosen generators such that $e(g_1, g_2) = g_t$.

We usually let the λ parameter be implicit, i.e. write $\mathbb{F}$ instead of $\mathbb{F}(\lambda)$. We write $\mathbb{G}_1$ and $\mathbb{G}_2$ additively. We use the notations $[x]_1 := x \cdot g_1$ and $[x]_2 := x \cdot g_2$.

We often denote by $[n]$ the integers $\{1, \ldots, n\}$. We use the acronym e.w.p for "except with probability"; i.e. e.w.p γ means with probability *at least* $1 - \gamma$.

universal SRS-based public-coin protocols We describe public-coin (meaning the verifier messages are uniformly chosen) interactive protocols between a prover and verifier; when deriving results for non-interactive protocols, we implicitly assume we can get a proof length equal to the total communication of the prover, using the Fiat-Shamir transform/a random oracle. Using this reduction between interactive and non-interactive protocols, we can refer to the "proof length" of an interactive protocol.

We allow our protocols to have access to a structured reference string (SRS) that can be derived in deterministic $\mathsf{poly}(\lambda)$-time from an "SRS of monomials" of the form $\left\{[x^i]_1\right\}_{a \le i \le b}$, $\left\{[x^i]_2\right\}_{c \le i \le d}$, for uniform $x \in \mathbb{F}$, and some integers a, b, c, d with absolute value bounded by $\mathsf{poly}(\lambda)$. It then follows from Bowe et al. [BGM17] that the required SRS can be derived in a universal and updatable setup requiring only one honest participant; in the sense that an adversary controlling all but one of the participants in the setup does not gain more than a $\mathsf{negl}(\lambda)$ advantage in its probability of producing a proof of any statement.

For notational simplicity, we sometimes use the SRS srs as an implicit parameter in protocols, and do not explicitly write it.

2.2 Analysis in the AGM model

For security analysis we will use the Algebraic Group Model of Fuchsbauer, Kiltz and Loss[FKL18]. In our protocols, by an *algebraic adversary* $\mathcal{A}$ in an SRS-based protocol we mean a $\mathsf{poly}(\lambda)$-time algorithm which satisfies the following.

- For $i \in \{1,2\}$, whenever $\mathcal{A}$ outputs an element $A \in \mathbb{G}_i$, it also outputs a vector v over $\mathbb{F}$ such that $A =< v, \mathsf{srs_i} >$.

Idealized verifier checks for algebraic adversaries We introduce some terminology to capture the advantage of analysis in the AGM.

First we say our srs *has degree* Q if all elements of $\mathsf{srs_i}$ are of the form $[f(x)]_i$ for $f \in \mathbb{F}_{<Q}[X]$ and uniform $x \in \mathbb{F}$. In the following discussion let us assume we are executing a protocol with a degree Q SRS, and denote by $f_{i,j}$ the corresponding polynomial for the j'th element of $\mathsf{srs_i}$.

Denote by a, b the vectors of $\mathbb{F}$-elements whose encodings in $\mathbb{G}_1, \mathbb{G}_2$ an algebraic adversary $\mathcal{A}$ outputs during a protocol execution; e.g., the j'th $\mathbb{G}_1$ element output by $\mathcal{A}$ is $[a_j]_1$.

By a "real pairing check" we mean a check of the form

$$(a \cdot T_1) \cdot (T_2 \cdot b) = 0$$

for some matrices T_1, T_2 over $\mathbb{F}$. Note that such a check can indeed be done efficiently given the encoded elements and the pairing function $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_t$.

Given such a "real pairing check", and the adversary $\mathcal{A}$ and protocol execution during which the elements were output, define the corresponding "ideal check" as follows. Since $\mathcal{A}$ is algebraic when he outputs $[a_j]_i$ he also outputs a vector v such that, from linearity, $a_j = \sum v_\ell f_{i,\ell}(x) = R_{i,j}(x)$ for $R_{i,j}(X) := \sum v_\ell f_{i,\ell}(X)$. Denote, for $i \in \{1,2\}$ the vector of polynomials $R_i = (R_{i,j})_j$. The corresponding ideal check, checks as a polynomial identity whether

$$(R_1 \cdot T_1) \cdot (T_2 \cdot R_2) \equiv 0$$

The following lemma is inspired by [FKL18]'s analysis of [Gro16], and tells us that for soundness analysis against algebraic adversaries it suffices to look at ideal checks. Before stating the lemma we define the Q-DLOG assumption similarly to [FKL18].

Definition 2.1. *Fix integer Q. The* Q-DLOG assumption for $(\mathbb{G}_1, \mathbb{G}_2)$ *states that given*

$$[1]_1, [x]_1, \ldots, \left[x^Q\right]_1, [1]_2, [x]_2, \ldots, \left[x^Q\right]_2$$

for uniformly chosen $x \in \mathbb{F}$, the probability of an efficient $\mathcal{A}$ outputting x is $\mathsf{negl}(\lambda)$.

The following lemma is proved in [GWC19] based on the arguments of [FKL18].

Lemma 2.2. *Assume the Q-DLOG for $(\mathbb{G}_1, \mathbb{G}_2)$. Given an algebraic adversary $\mathcal{A}$ participating in a protocol with a degree Q SRS, the probability of any real pairing check passing is larger by at most an additive* $\mathsf{negl}(\lambda)$ *factor than the probability the corresponding ideal check holds.*

Knowlege soundness in the Algebraic Group Model We say a protocol $\mathscr{P}$ between a prover $\mathbf{P}$ and verifier $\mathbf{V}$ for a relation $\mathcal{R}$ has *Knowledge Soundness in the Algebraic Group Model* if there exists an efficient E such that the probability of any algebraic adversary $\mathcal{A}$ winning the following game is $\mathsf{negl}(\lambda)$.

1. $\mathcal{A}$ chooses input x and plays the role of $\mathbf{P}$ in $\mathscr{P}$ with input x.
2. E given access to all of $\mathcal{A}$'s messages during the protocol (including the coefficients of the linear combinations) outputs ω.
3. $\mathcal{A}$ wins if
   (a) $\mathbf{V}$ outputs acc at the end of the protocol, and
   (b) $(\mathsf{x}, \omega) \notin \mathcal{R}$.

2.3 Polynomial commitment schemes

We define polynomial commitment schemes similarly to [GWC19]. Specifically, we define the open procedure in a batched setting having multiple polynomials and evaluation points.

Definition 2.3. *A d-polynomial commitment scheme is a triplet* $\mathscr{S} = (\mathsf{gen}, \mathsf{com}, \mathsf{open})$ *such that*

- $\mathsf{gen}(d)$ *- is a randomized algorithm that given positive integer d outputs a structured reference string (SRS)* srs.
- $\mathsf{com}(f, \mathsf{srs})$ *- is an algorithm that given a polynomial* $f \in \mathbb{F}_{<d}[X]$ *and an output* srs *of* $\mathsf{gen}(d)$ *returns a commitment* cm *to* f.
- open *is a public coin protocol between parties* P_{PC} *and* V_{PC}. P_{PC} *is given* $f_1, \ldots, f_k \in \mathbb{F}_{<d}[X]$. P_{PC} *and* V_{PC} *are both given*
  1. *positive integers* $d, t = \mathsf{poly}(\lambda)$,
  2. $\mathsf{srs} = \mathsf{gen}(d)$,
  3. *a subset* $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}$,
  4. *subsets* $S_1, \ldots, S_k \subset T$,
  5. $\mathsf{cm}_1, \ldots, \mathsf{cm}_k$ *- the alleged commitments to* $f_1, \ldots, f_k$,
  6. $\{s_{i,z}\}_{i \in [k], z \in S_i}$ *- the alleged correct openings* $\{f_i(z)\}_{i \in [k], z \in S_i}$.

  At the end of the protocol V_{PC} *outputs* acc *or* rej*; such that*
  - ***Completeness:*** *Fix any* $k, t = \mathsf{poly}(\lambda)$, $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}$, $S_1, \ldots, S_k \subset T$, $f_1, \ldots, f_k \in \mathbb{F}_{<d}[X]$. *Suppose that for each* $i \in [k]$, $\mathsf{cm}_i = \mathsf{com}(f_i, \mathsf{srs})$, *and for each* $i \in [k]$ *and* $z \in S_i$ *we have* $s_{i,z} = f_i(z)$. *Then if* P_{PC} *follows* open *correctly with these values,* V_{PC} *outputs* acc *with probability one.*

– ***Knowledge soundness in the algebraic group model:*** *There exists an efficient E such that for any algebraic adversary $\mathcal{A}$ and any choice of $d = \mathsf{poly}(\lambda)$ the probability of $\mathcal{A}$ winning the following game is $\mathsf{negl}(\lambda)$ over the randomness of $\mathcal{A}$ and* gen.

1. *Given d and $\mathsf{srs} = \mathsf{gen}(d)$, $\mathcal{A}$ outputs $\mathsf{cm}_1, \ldots, \mathsf{cm}_k \in \mathbb{G}_1$.*
2. *E, given access to the messages of $\mathcal{A}$ during the previous step, outputs $f_1, \ldots, f_k \in \mathbb{F}_{<d}[X]$.*
3. *$\mathcal{A}$ outputs $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}$, $S_1, \ldots, S_k \subset T$, $\mathbb{F}$, $\{s_{i,z}\}_{i\in[k], z\in S_i}$.*
4. *$\mathcal{A}$ takes the part of P_{PC} in the protocol open with the inputs $\mathsf{cm}_1, \ldots, \mathsf{cm}_k$,$T, S_1, \ldots, S_k$, $\{s_{i,z}\}$.*
5. *$\mathcal{A}$ wins if*
   * *V_{PC} outputs acc at the end of the protocol.*
   * *For some $i \in [k]$ and $z \in S_i$, $s_{i,z} \neq f_i(z)$.*

3 The new scheme

We first state the following straightforward claim that allows us to efficiently "uniformize" checks on different evaluation points.

Claim 3.1. *Fix subsets $S \subset T \subset \mathbb{F}$, and polynomials $f, r \in \mathbb{F}_{<d}[X]$. Then $f(z) = r(z)$ for each $z \in S$ if and only if Z_T divides $Z_{T\setminus S} \cdot (f(X) - r(X))$.*

Proof. Z_T divides $Z_{T\setminus S} \cdot (f(X) - r(X))$ if and only if $Z_{T\setminus S} \cdot (f(X) - r(X))$ vanishes on T if and only if $(f(X) - r(X))$ vanishes on S. □

We also use the following claim, which is part of Claim 4.6 in [GWC19] where a proof of it can be found.

Claim 3.2. *Fix $F_1, \ldots, F_k \in \mathbb{F}_{<n}[X]$. Fix $Z \in \mathbb{F}_{<n}[X]$ that decomposes to distinct linear factors over $\mathbb{F}$. Suppose that for some $i \in [k]$, $Z \nmid F_i$. Then, e.w.p $k/|\mathbb{F}|$ over uniform $a \in \mathbb{F}$, Z doesn't divide*

$$G := \sum_{j=1}^{k} a^j \cdot F_j.$$

We are ready to describe our PCS.

1. $\mathsf{gen}(d)$ - choose uniform $x \in \mathbb{F}$. Output $\mathsf{srs} = \left([1]_1, [x]_1, \ldots, \left[x^{d-1}\right]_1, [1]_2, [x]_2, \ldots, \left[x^t\right]_2\right)$.
2. $\mathsf{com}(f, \mathsf{srs}) := [f(x)]_1$.
3. $\mathsf{open}\left(d, t, \{\mathsf{cm}_i\}_{i\in[k]}, T = \{z_1, \ldots, z_t\} \subset \mathbb{F}, \{S_i \subset T\}_{i\in[k]}, \{s_{i,z}\}_{i\in[k], z\in S_i}\right)$:
   (a) V_{PC} sends random $\gamma \in \mathbb{F}$.

(b) For $i \in [k]$, $\mathsf{V_{PC}}$ and $\mathsf{P_{PC}}$ both compute the polynomials $\{r_i\}_{i\in[k]}$ such that $r_i \in \mathbb{F}_{<|S_i|}[X]$ satisfies $r_i(z) = s_{i,z}$ for each $z \in S_i$.

(c) $\mathsf{P_{PC}}$ computes the polynomial

$$h(X) := \sum_{i=1}^{t} \gamma^{i-1} \cdot \frac{f_i(X) - r_i(X)}{Z_{S_i(X)}}$$

and using srs computes and sends $W := [h(x)]_1$.

(d) $\mathsf{V_{PC}}$ computes for each $i \in [k]$, $Z_i := \left[Z_{T\setminus S_i}(x)\right]_2$.

(e) $\mathsf{V_{PC}}$ computes

$$F := \sum_{i\in[k]} \gamma^{i-1} \cdot e(\mathsf{cm}_i - [r_i(x)]_1, Z_i).$$

(f) $\mathsf{V_{PC}}$ outputs acc if and only if

$$F = e(W, [Z_T(x)]_2).$$

We argue knowledge soundness for the above protocol. More precisely, we argue the existence of an efficient E such that an algebraic adversary $\mathcal{A}$ can only win the KS game w.p. $\mathsf{negl}(\lambda)$.

Let $\mathcal{A}$ be such an algebraic adversary.

$\mathcal{A}$ begins by outputting $\mathsf{cm}_1, \ldots, \mathsf{cm}_k \in \mathbb{G}_1$. Each cm_i is a linear combination $\sum_{j=0}^{d-1} a_{i,j} \left[x^j\right]_1$. E, who is given the coefficients $\{a_{i,j}\}$, simply outputs the polynomials

$$f_i(X) := \sum_{j=0}^{d-1} a_{i,j} \cdot X^j.$$

$\mathcal{A}$ now outputs $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}, \{S_i \subset T\}_{i\in[k]}, \{s_{i,z}\}_{i\in[k], z\in S_i}$. Define, for each $i \in [k]$, $r_i \in \mathbb{F}_{<|S_i|}[X]$ such that $r_i(z) = s_{i,z}$ for each $z \in S_i$. Assume that for some $i^* \in [k], z^* \in S_{i^*}$, we have $f_{i^*}(z^*) \neq r_{i^*}(z^*)$. We show that for any strategy of $\mathcal{A}$ from this point, $\mathsf{V_{poly}}$ outputs acc w.p. $\mathsf{negl}(\lambda)$.

In the first step of open, $\mathsf{V_{poly}}$ chooses a random $\gamma \in \mathbb{F}$. Let

$$f(X) := \sum_{i\in[t]} \gamma^{i-1} \cdot Z_{T\setminus S_i} \cdot (f_i(X) - r_i(X)).$$

We know from Claim 3.1 that $F_{i^*} := Z_{T\setminus S_{i^*}} \cdot (f_{i^*}(X) - r_{i^*}(X))$ isn't divisible by Z_T. Thus, using Claim 3.2, we know that e.w.p $k/|\mathbb{F}|$ over γ, f isn't divisble by Z_T. Now $\mathcal{A}$ outputs $W = [H(x)]_1$ for some $H \in \mathbb{F}_{<d}[X]$. According to Lemma 2.2, it suffices to upper bound the probability that the ideal check corresponding to the real pairing check in the protocol passes. It has the form

$$f(X) \equiv H(X)Z_T(X).$$

The check passing implies that $f(X)$ is divisible by Z_T. Thus the ideal check can only pass w.p. $k/|\mathbb{F}| = \mathsf{negl}(\lambda)$ over the randomness of $\mathrm{V}_{\mathsf{poly}}$, which implies the same thing for the real check according to Lemma 2.2.

We summarize the efficiency properties of the scheme.

Lemma 3.3. *Fix positive integer d. There is a d-polynomial commitment scheme $\mathscr{S}$ such that*

1. *For $n \leq d$ and $f \in \mathbb{F}_{<n}[X]$, computing $\mathsf{com}(f)$ requires n $\mathbb{G}_1$-exponentiations.*
2. *Given $T := (z_1, \ldots, z_t) \in \mathbb{F}^t, f_1, \ldots, f_k \in \mathbb{F}_{<d}[X], \{S_i\}_{i\in[k]}$, denote by k^* the number of distinct subsets $\{S_1^*, \ldots, S_{k^*}^*\}$ in $\{S_i\}$; and let $K := t + \sum_{i\in[k^*]} (t - |S_i^*|)$. and denote $d^* := \max\{\deg(f_i)\}_{i\in[k]}$. Let $\mathsf{cm}_i = \mathsf{com}(f_i)$. Then open $(\{cm_i\},\{f_i\},T,\{S_i \subset T\},\{s_{i,z}\})$ requires*
   (a) *A single $\mathbb{G}_1$ element to be passed from $\mathrm{P}_{\mathsf{poly}}$ to $\mathrm{V}_{\mathsf{poly}}$.*
   (b) *At most d^* $\mathbb{G}_1$-exponentiations of $\mathrm{P}_{\mathsf{poly}}$.*
   (c) *$k-1$ $\mathbb{G}_1$-exponentiations, K $\mathbb{G}_2$-exponentiations and k^*+1 pairings of $\mathrm{V}_{\mathsf{poly}}$.*

4 Reducing verifier operations at the expense of proof length

We describe a variant of the scheme of Section 3 where we eliminate the verifier's $\mathbb{G}_2$ operations and reduce the number of pairings to two. This comes at the cost of an extra $\mathbb{G}_1$ element sent by the prover. We state the result first, followed by the description of the scheme.

Lemma 4.1. *Fix positive integer d. There is a d-polynomial commitment scheme $\mathscr{S}$ such that*

1. *For $n \leq d$ and $f \in \mathbb{F}_{<n}[X]$, computing $\mathsf{com}(f)$ requires n $\mathbb{G}_1$-exponentiations.*
2. *Given $T := (z_1, \ldots, z_t) \in \mathbb{F}^t, f_1, \ldots, f_k \in \mathbb{F}_{<d}[X], \{S_i\}_{i\in[k]}$, denote by k^* the number of distinct subsets $\{S_1^*, \ldots, S_{k^*}^*\}$ in $\{S_i\}$; and let $K := t + 1 + \sum_{i\in[k]} (|S_i|)$. and denote $d^* := \max\{\deg(f_i)\}_{i\in[k]}$. Let $\mathsf{cm}_i = \mathsf{com}(f_i)$. Then open $(\{cm_i\},\{f_i\},T,\{S_i \subset T\},\{s_{i,z}\})$ requires*
   (a) *Two $\mathbb{G}_1$ elements to be passed from P_{PC} to V_{PC}.*
   (b) *At most $2d^* + 1$ $\mathbb{G}_1$-exponentiations of P_{PC}.*
   (c) *K $\mathbb{G}_1$-exponentiations, and two pairings of V_{PC}.*

1. $\mathsf{gen}(d)$ - choose uniform $x \in \mathbb{F}$. Output $\mathsf{srs} = \left([1]_1, [x]_1, \ldots, \left[x^{d-1}\right]_1, [1]_2, [x]_2\right)$.
2. $\mathsf{com}(f, \mathsf{srs}) := [f(x)]_1$.
3. $\mathsf{open}\left(d, t, \{\mathsf{cm}_i\}_{i\in[k]}, T = \{z_1, \ldots, z_t\} \subset \mathbb{F}, \{S_i \subset T\}_{i\in[k]}, \{s_{i,z}\}_{i\in[k], z\in S_i}\right)$:

(a) V_{PC} sends random $\gamma \in \mathbb{F}$.

(b) For $i \in [k]$, V_{PC} and P_{PC} both compute the polynomials $\{r_i\}_{i\in[k]}$ such that $r_i \in \mathbb{F}_{<|S_i|}[X]$ satisfies $r_i(z) = s_{i,z}$ for each $z \in S_i$.

(c) P_{PC} computes the polynomial

$$f(X) := \sum_{i=1}^{t} \gamma^{i-1} \cdot Z_{T\setminus S_i}(X) \cdot (f_i(X) - r_i(X)).$$

Recall that f is divisible by Z_T according to Claim 3.2, and define $h(X) := f(X)/Z_T(X)$. Using srs, P_{PC} computes and sends $W := [h(x)]_1$.

(d) V_{PC} sends random $z \in \mathbb{F}$.

(e) P_{PC} computes the polynomial

$$L(X) := f_z(X) - Z_T(z) \cdot h(X),$$

where

$$f_z(X) := \sum_{i=1}^{t} \gamma^{i-1} \cdot Z_{T\setminus S_i}(z) \cdot (f_i(X) - r_i(X))$$

Note that $L(z) = f(z) - Z_T(z) \cdot h(z) = 0$, and thus $(X - z)$ divides L. P_{PC} sends $W' := \left[\frac{L(x)}{x-z}\right]_1$.

(f) V_{PC} computes:

$$F := \sum_{i\in[t]} \gamma^{i-1} \cdot Z_{T\setminus S_i}(z) \cdot (\mathsf{com}(f_i) - [r_i(X)]_1) - Z_T(z) \cdot W$$

(g) V_{PC} outputs acc if and only if

$$e(F, [1]_2) = e(W', [x - z]_2).$$

We argue knowledge soundness for the above protocol.More precisely, we argue the existence of an efficient E such that an algebraic adversary $\mathcal{A}$ can only win the KS game w.p. $\mathsf{negl}(\lambda)$. The proof begins identically to the previous one.

Let $\mathcal{A}$ be such an algebraic adversary.

$\mathcal{A}$ begins by outputting $\mathsf{cm}_1, \ldots, \mathsf{cm}_k \in \mathbb{G}_1$. Each cm_i is a linear combination $\sum_{j=0}^{d-1} a_{i,j} \left[x^j\right]_1$. E, who is given the coefficients $\{a_{i,j}\}$, simply outputs the polynomials

$$f_i(X) := \sum_{j=0}^{d-1} a_{i,j} \cdot X^j.$$

$\mathcal{A}$ now outputs $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}, \{S_i \subset T\}_{i\in[k]}, \{s_{i,z}\}_{i\in[k],z\in S_i}$. Define, for each $i \in [k]$, $r_i \in \mathbb{F}_{<|S_i|}[X]$ such that $r_i(z) = s_{i,z}$ for each $z \in S_i$. Assume that for some

$i^* \in [k], z^* \in S_{i^*}$, we have $f_{i^*}(z^*) \neq r_{i^*}(z^*)$. We show that for any strategy of $\mathcal{A}$ from this point, $\mathrm{V}_{\mathsf{poly}}$ outputs acc w.p. $\mathsf{negl}(\lambda)$.

In the first step of open, $\mathrm{V}_{\mathsf{poly}}$ chooses a random $\gamma \in \mathbb{F}$. Let

$$f(X) := \sum_{i \in [t]} \gamma^{i-1} \cdot Z_{T \setminus S_i} \cdot (f_i(X) - r_i(X)).$$

We know from Claim 3.1 that $F_{i^*} := Z_{T \setminus S_{i^*}} \cdot (f_{i^*}(X) - r_{i^*}(X))$ isn't divisible by Z_T. Thus, using Claim 3.2, we know that e.w.p $k/|\mathbb{F}|$ over γ, f isn't divisble by Z_T. Assume we are in this case. Now $\mathcal{A}$ outputs $W = [H(x)]_1$ for some $H \in \mathbb{F}_{<d}[X]$, followed by V_{PC} sending uniform $z \in \mathbb{F}$. Since we are in the case that f isn't divisble by Z_T, we know there are at most $2d$ values $z \in \mathbb{F}$ such that $f(z) = H(z) \cdot Z_T(z)$; and thus z chosen by V_{PC} is of this form only w.p. $\mathsf{negl}(\lambda)$. Assume z sent by V_{PC} is not of this form. P_{PC} now outputs $W' = [H'(x)]_1$ for some $H' \in \mathbb{F}_{<d}[X]$ According to Lemma 2.2, it suffices to upper bound the probability that the ideal check corresponding to the real pairing check in the protocol passes. Denoting

$$L'(X) := \sum_{i \in [t]} Z_{T \setminus S_i}(z) \cdot (f_i(X) - r_i(X)) - Z_T(z) \cdot H(X),$$

the ideal check has the form

$$L'(X) \equiv H'(X) \cdot (X - z),$$

and thus can pass for some $H' \in \mathbb{F}_{<d}[X]$ only if L' is divisible by $(X - z)$, which means $L'(z) = 0$. However

$$L'(z) = \sum_{i \in [t]} Z_{T \setminus S_i}(z) \cdot (f_i(z) - r_i(z)) - Z_T(z) \cdot H(z) = f(z) - Z_T(z) \cdot H(z),$$

and we are in the case where $f(z) \neq Z_T(z) \cdot H(z)$. In summary, the ideal check can only pass w.p. $\mathsf{negl}(\lambda)$ over the randomness of V_{PC}, which implies the same thing for the real check according to Lemma 2.2.

Acknowledgements

Part of this research was conducted while the second author was supported by Protocol Labs.

References

[BCI+13] N. Bitansky, A. Chiesa, Y. Ishai, R. Ostrovsky, and O. Paneth. Succinct non-interactive arguments via linear interactive proofs. In *Theory of Cryptography - 10th Theory of Cryptography Conference, TCC 2013, Tokyo, Japan, March 3-6, 2013. Proceedings*, pages 315–333, 2013.

[BFS19] B. Bünz, B. Fisch, and A. Szepieniec. Transparent snarks from DARK compilers. *IACR Cryptology ePrint Archive*, 2019:1229, 2019.

[BGM17] S. Bowe, A. Gabizon, and I. Miers. Scalable multi-party computation for zk-snark parameters in the random beacon model. Cryptology ePrint Archive, Report 2017/1050, 2017. `https://eprint.iacr.org/2017/1050`.

[CHM+19] A. Chiesa, Y. Hu, M. Maller, P. Mishra, N. Vesely, and N. P. Ward. Marlin: Preprocessing zksnarks with universal and updatable SRS. *IACR Cryptology ePrint Archive*, 2019:1047, 2019.

[FKL18] G. Fuchsbauer, E. Kiltz, and J. Loss. The algebraic group model and its applications. In *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II*, pages 33–62, 2018.

[Gab19] A. Gabizon. Auroralight:improved prover efficiency and SRS size in a sonic-like system. *IACR Cryptology ePrint Archive*, 2019:601, 2019.

[Gro16] J. Groth. On the size of pairing-based non-interactive arguments. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II*, pages 305–326, 2016.

[GWC19] A. Gabizon, Z. J. Williamson, and O. Ciobotaru. PLONK: permutations over lagrange-bases for oecumenical noninteractive arguments of knowledge. *IACR Cryptology ePrint Archive*, 2019:953, 2019.

[KZG10] A. Kate, G. M. Zaverucha, and I. Goldberg. Constant-size commitments to polynomials and their applications. pages 177–194, 2010.

[MBKM19] M. Maller, S. Bowe, M. Kohlweiss, and S. Meiklejohn. Sonic: Zero-knowledge snarks from linear-size universal and updateable structured reference strings. *IACR Cryptology ePrint Archive*, 2019:99, 2019.

[PHGR16] B. Parno, J. Howell, C. Gentry, and M. Raykova. Pinocchio: nearly practical verifiable computation. *Commun. ACM*, 59(2):103–112, 2016.