PLONK, SHPLONK

Justin Drake
Ethereum Foundation

Ariel Gabizon
Protocol Labs

Zachary J. Williamson
Aztec Protocol

December 11, 2019

Abstract

We present an enhanced version of the Kate, Zaverucha, Goldberg polynomial commitment scheme [KZG10] where a single group element can be an openning proof for multiple polynomials at multiple points. We apply this to the **Plonk** proof system to obtain "for free" constraints involving multiple shifts of a witness.

1 Introduction

1.1 Our results

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 $\left\{\left[x^i\right]_1\right\}_{a\leq i\leq b}$, $\left\{\left[x^i\right]_2\right\}_{c\leq i\leq d}$, 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 $e:\mathbb{G}_1\times\mathbb{G}_2\rightarrow\mathbb{G}_t$.

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*

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

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

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.*

Proof. Let γ be the difference between the satisfiability of the real and ideal check. We describe an adversary $\mathcal{A}^*$ for the Q-DLOG problem that succeeds with probability γ; this implies $\gamma = \mathsf{negl}(\lambda)$. $\mathcal{A}^*$ receives the challenge

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

and constructs using group operations the correct SRS for the protocol. Now $\mathcal{A}^*$ runs the protocol with $\mathcal{A}$, simulating the verifier role. Note that as $\mathcal{A}^*$ receives from $\mathcal{A}$ the vectors of coefficients v, he can compute the polynomials $\{R_{i,j}\}$ and check if we are in the case that the real check passed but ideal check failed. In case we are in this event, $\mathcal{A}^*$ computes

$$R := (R_1 \cdot T_1)(T_2 \cdot R_2).$$

We have that $R \in \mathbb{F}_{<2Q}[X]$ is a non-zero polynomial for which $R(x) = 0$. Thus $\mathcal{A}^*$ can factor R and find x. □

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) $(\mathsf{x}, \omega) \notin \mathcal{R}$.

3 A batched version of the [KZG10] scheme

Crucial to the efficiency of our protocol is a batched version of the [KZG10] polynomial commitment scheme similar to Appendix C of [MBKM19], allowing to query multiple committed polynomials at multiple points. We begin by defining polynomial commitment schemes in a manner conducive to our protocol. Specifically, we define the open procedure in a batched setting having multiple polynomials and evaluation points.

Definition 3.1. *A d-polynomial commitment scheme consists of*

- $\mathsf{gen}(d)$ - *a randomized algorithm that outputs an SRS* srs.
- $\mathsf{com}(f, \mathsf{srs})$ - *that given a polynomial* $f \in \mathbb{F}_{<d}[X]$ *returns a commitment* cm *to* f.
- *A public coin protocol* open *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* $t = \mathsf{poly}(\lambda)$, *a subset* $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}$, *subsets* $S_1, \ldots, S_k \subset T$, $\mathsf{cm}_1, \ldots, \mathsf{cm}_k$ *- the alleged commitments to* $f_1, \ldots, f_k$, $\{s_{i,z}\}_{i\in[k], z\in S_i}$ *- 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* $k, t = \mathsf{poly}(\lambda)$, $T = \{z_1, \ldots, z_t\} \subset \mathbb{F}$, $S_1, \ldots, S_k \subset T$, $f_1, \ldots, f_k \in polysofdegd$. *Suppose that for each* $i \in [k]$, $\mathsf{cm}_i = \mathsf{com}(f_i, \mathsf{srs})$, *and for each* $i \in [k], z \in S_i, s_{i,z} = f_i(z)$. *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* $\mathcal{A}$ *the probability of* $\mathcal{A}$ *winning the following game is* $\mathsf{negl}(\lambda)$ *over the randomness of* $\mathcal{A}$ *and* gen.
  1. *Given* srs, $\mathcal{A}$ *outputs* $t, \mathsf{cm}_1, \ldots, \mathsf{cm}_k \in \mathbb{G}_1$.
  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* $i \in [k], z \in S_i$, $s_{i,z} \neq f_i(z)$.

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

Claim 3.2. *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. $\square$

We describe the following scheme based on [KZG10].

1. $\mathsf{gen}(d)$ - choose uniform $x \in \mathbb{F}$. Output $\mathsf{srs} = ([1]_1, [x]_1, \ldots, [x^{d-1}]_1, [1]_2, [x]_2, \ldots, [x^t]_2)$.
2. $\mathsf{com}(f, \mathsf{srs}) := [f(x)]_1$.
3. For $i \in [k]$,
   $\mathsf{open}(\{\mathsf{cm}_i\}, T = \{z_1, \ldots, z_t\} \subset \mathbb{F}, \{S_i \subset T\}_{i\in[k]}, \{s_{i,z}\}_{i\in[k], z\in S_i})$:
   (a) V_{PC} sends random $\gamma \in \mathbb{F}$.
   (b) 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
   $$h(X) := \sum_{i=1}^{t} \gamma^{i-1} \cdot \frac{f_i(X) - r_i(X)}{Z_{S_i(X)}}$$
   and using srs computes and sends $W := [h(x)]_1$.
   (d) V_{PC} computes for each $i \in [k]$, $Z_i := [Z_{T\setminus S_i}]_2$.
   (e) V_{PC} computes
   $$F := \sum_{i\in[k]} \gamma^{i-1} \cdot e(\mathsf{cm}_i - [r_i(x)]_1, Z_i).$$
   (f) 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 $\sum_{j=0}^{d-1} a_{i,j} [x^j]_1$. 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 $i' \in [k], z' \in S_i$, $f_{i'}(z') \neq r_{i'}(z') = s_{i',z'}$. 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 3.2 that $F_{i'} := Z_{T \setminus S_{i'}} \cdot (f_{i'}(X) - r_{i'}(X))$ isn't divisible by Z_T. Thus using the derandomized version of Claim 4.6 from **Plonk**, 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 $K := t + 1 + \sum_{i \in [k^*]} t - |S_i^*| + 1$. 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,$\{S_i\}$,$\{s_{i,z}\}$) requires*
   - *(a) A single $\mathbb{G}_1$ element to be passed from* $\mathrm{P}_{\mathsf{poly}}$ *to* $\mathrm{V}_{\mathsf{poly}}$.
   - *(b) At most $d^* + 1$ $\mathbb{G}_1$-exponentiations of* $\mathrm{P}_{\mathsf{poly}}$.
   - *(c) k $\mathbb{G}_1$-exponentiations, K $\mathbb{G}_2$-exponentiations and $k^* + 1$ pairings of* $\mathrm{V}_{\mathsf{poly}}$.

References

[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`.

[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.

[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.

[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.