On Black-Box Knowledge-Sound Commit-And-Prove SNARKs

Wednesday 20th September, 2023, 12:21

Helger Lipmaa[0000−0001−8393−6821]

University of Tartu, Tartu, Estonia

Abstract. Gentry and Wichs proved that adaptively sound SNARGs for hard languages need non-falsifiable assumptions. Lipmaa and Pavlyk claimed Gentry-Wichs is tight by constructing a non-adaptively sound zk-SNARG FANA for NP from falsifiable assumptions. We show that FANA is flawed. We define and construct a fully algebraic F-position-binding vector commitment scheme VCF. We construct a concretely efficient commit-and-prove zk-SNARK Punic, a version of FANA with an additional VCF commitment to the witness. Punic satisfies *semi-adaptive black-box G-knowledge-soundness*, a new natural knowledge-soundness notion for commit-and-prove SNARKs. We use a new proof technique to achieve global consistency using a functional somewhere-extractable commitment scheme to extract vector commitment's local proofs.

Keywords: Commit-and-prove, falsifiable assumptions, Gentry-Wichs, non-adaptive soundness, QA-NIZK, vector commitment, zk-SNARK

1 Introduction

Gentry and Wichs [26] proved non-falsifiable assumptions are needed to construct (even non-zero-knowledge) adaptively sound SNARGs (*succinct non-interactive arguments*, [30,49,50,25,56,31]) for hard languages under black-box reductions. Their impossibility result balances the following four properties of NIZKs: (1) *Succinctness*: Non-succinct NIZKs are not suitable for many applications. (2) *Falsifiability:* an assumption or a primitive is *falsifiable* if one can efficiently decide whether the adversary broke it. Non-falsifiable assumptions are highly controversial. (3) *Adaptive soundness*: the SNARG is sound even if the malicious prover can choose the statement $\mathtt{x}$ after seeing the CRS. Non-adaptive soundness guarantees security only if $\mathtt{x}$ is independent of the CRS. (4) Many applications need SNARGs for *hard languages* (i.e., languages with hard subset membership problem) like circuit satisfiability.

Assuming black-box reductions, Gentry-Wichs is known to be tight in three aspects: (1) non-succinct falsifiable assumption-based adaptively sound NIZKs are known for NP [20], (2) falsifiable assumption-based adaptively sound SNARGs are known for P [41], and (3) non-falsifiable assumption-based adaptively sound zk-SNARKs are known for NP [30].

It has been a major open problem *whether Gentry-Wichs is tight in the fourth aspect; that is, whether falsifiable assumption-based non-adaptively sound (even non-zero knowledge) SNARGs for hard languages exist.* Intuitively, it is easier to achieve non-adaptive than adaptive black-box knowledge soundness since, in the former case, the extractor has additional power. Namely, it can rewind the prover to the point after the prover chose the statement, sample a new CRS, and thus obtain many arguments of the same statement under different CRSs.

Sahai and Waters built a non-adaptively sound zk-SNARG for NP [58] using iO, one-way functions, and succinct punctured PRFs. One can use sub-exponential but falsifiable assumptions to instantiate iO [38]. However, their SNARG has exponential security loss in witness length [39]. Since the reduction can decide the language, their SNARG bypasses the Gentry-Wichs impossibility result (and can achieve adaptive security by complexity leveraging, [39]). Hence, constructing non-adaptively sound SNARGs for NP remains open after [58]. Jain and Jin [39] proposed a SNARG that overcomes this limitation, but only for a subclass of languages in $\mathsf{NP} \cap \mathsf{co\text{-}NP}$ with a "PV proof of disjointness".

Lipmaa and Pavlyk [52] proposed FANA, an *efficient* (and *polynomial-time challenger*) falsifiable assumption-based non-adaptively sound zk-SNARG for NP. FANA is based on two earlier constructions, DGPRS of [18] and FLPS [19]. DGPRS and FLPS are adaptively sound commit-and-prove (C&P) SNARGs for NP. Since they have non-succinct commitments, Gentry-Wichs does not apply.

By leveraging continuous leakage-resilient one-way functions (that exist under the discrete logarithm assumption [4]), Campanelli et al. [12] proved that non-adaptive black-box extractable SNARKs (*succinct non-interactive arguments of knowledge*, i.e., *knowledge*-sound arguments) for NP do not exist. Recall that extraction is *black-box* if it extracts a witness from a prover only using its input/output interface, without knowledge about its internal state or code. Note that [12] does not contradict [52] who construct a SNARG.

Our First Contribution. We show that FANA's security proof is flawed, and FANA is *not* a non-adaptively SNARG.[1] The main reason why FANA's proof breaks down is that, differently from DGPRS and FLPS, FANA is *not* a C&P SNARG. On the other hand, DGPRS and FLPS rely on a perfectly binding (non-succinct) commitment scheme, i.e., they are not SNARGs.

Main Question. In Table 1, we summarize the state of the art: on top of [26], [12] proved that falsifiable assumption-based non-adaptively knowledge-sound SNARKs for NP do not exist, while [52] (that is insecure) and [58] (with an exponential security loss) constructed falsifiable assumption-based non-adaptively sound SNARGs for NP. This leaves two open questions: Can one construct falsifiable assumption-based (1) non-adaptively sound SNARGs, and (2) SNARKs for NP under a different adaptivity notion?

We do not know how to answer (1), i.e., formally settle the tightness of Gentry-Wichs. Instead, the current paper aims to find a solution in the latter

[1] [59] noted that FANA is insecure (and referred to a private conversation with the authors of [52]), but they did not explain why. We will provide full details.

Table 1. The known possibility and impossibility results for falsifiable assumption-based SNAR(G|K)s for hard languages.

Adaptivity/Knowledge	SNARG	SNARK
Adaptive	✗ [26]	✗ [26]
Non-adaptive	✓/[52] (✓ with exp. security loss [58])	✗ [12]
Semi-adaptive	✓ This work	✓ This work

direction. There, one has the following natural question: For what notion of adaptivity can one construct falsifiable assumption-based black-box knowledge-sound SNARKs for NP? Moreover, can this be done *efficiently*?

Our (Four More) Contributions. Second, definition. We define *semi-adaptive black-box knowledge-soundness*, a natural security notion for falsifiable assumption-based C&P SNARKs. In a black-box knowledge-sound C&P SNARK, one can black-box extract partial witnesses by rerunning the adversary on a fixed commitment key and commitment C (to the witness) but many CRSs. One can recover the full witness from many succinct arguments and thus overcome an information-theoretical barrier plaguing SNARKs. This is similar to using rewinding in interactive zero-knowledge proofs; indeed, the definition is related to that of witness-extended emulation [48]. Crucially, having a C&P SNARK (i.e., a fixed commitment key and a commitment) lets us avoid the impossibility result of [12]. We emphasize that finding a correct definition is one of the most critical tasks in cryptographic research.

Third, modular proof. We prove black-box knowledge-soundness in two steps, as standard in the *interactive* arguments but unlike [18,19,52]. First, we define *semi-adaptive computational special soundness*, a variant of special soundness [16]. We prove that every semi-adaptively computationally special sound and CRS-indistinguishable C&P zk-SNARK is also semi-adaptively black-box knowledge sound. Thus, we only need to prove the former two properties.

Fourth, the proof technique. We use a perfectly hiding *vector* commitment scheme VC [47,36,13] to create C. Since VC is perfectly hiding, one cannot black-box extract from C. Instead, we use a functional somewhere-extractable (FSE) commitment scheme [19] to black-box extract a *partial witness* (VC's local opening and proof) from a FSE commitment. We then combine many partial witnesses into a full witness. We define and construct *fully algebraic F-position-binding vector commitment schemes* that allow such extractions.

Fifth, construction. We construct a C&P zk-SNARK Punic that fixes FANA by (re)adding a language parameter $\mathsf{lp} = \mathsf{ck}$ and a succinct (in our case, *vector*) commitment C to $(\mathbb{x}, \mathbb{w})$. We prove Punic is semi-adaptively computationally special-sound and CRS-indistinguishable and thus semi-adaptively black-box knowledge-sound. Since one of our primary goals is efficiency, the special soundness of Punic is based on non-standard yet non-interactive and known falsifiable assumptions.

On Tightness of Gentry-Wichs. The current work opens a novel approach to studying the tightness of Gentry-Wichs in the context of C&P SNARKs. Table 1 summarizes the known results. We emphasize that it is unknown whether one can construct falsifiable assumption-based non-adaptively sound SNARGs for NP with polynomial security loss. We leave it as the open question to state a precise version of Gentry-Wichs for both C&P and non-C&P SNARGs and SNARKs. In particular, is there a separation between SNARGs and black-box knowledge-sound SNARKs?

2 Technical Overview

We will start this section with an overview of DGPRS, FLPS, and FANA. After that, we describe our contributions in more detail.

2.1 Background

In C&P SNARGs and SNARKs, the CRS includes a commitment key ($\Gamma.\mathsf{ck}$, also called a *language parameter* lp, [40]), and the statement includes a Γ-commitment C. Here, Γ is an extractable commitment scheme. Most of the efficient SNARKs (e.g., [25,56,31,14,11,53]) are C&P SNARKs although usually not stated as such; in their knowledge-soundness proof, one uses knowledge assumptions to *non-black-box* extract the full witness from C. Different definitions of C&P SNARGs allow or do not allow dependencies between the commitment key, the language, and the CRS. The definition of C&P QA-SNARG (quasi-adaptive SNARG[2], [40,18]) explicitly requires that one first fixes $\mathsf{lp} = \Gamma.\mathsf{ck}$, defining (for some relation $\mathcal{R}$) the language

$$\mathcal{L}_{\mathsf{lp}} = \{(C, \mathbb{x}) : (\exists \mathbb{w}, r)\, (C = \Gamma.\mathsf{Com}(\Gamma.\mathsf{ck}, (\mathbb{x}, \mathbb{w}); r) \wedge (\mathbb{x}, \mathbb{w}) \in \mathcal{R})\} \ ,$$

then a CRS crs that may depend on lp (and thus $\mathcal{L}_{\mathsf{lp}}$). Only after that does the prover choose a statement $(C, \mathbb{x})$. Quasi-adaptive soundness is defined for this temporal order: for any honestly generated lp (that fixes $\mathcal{L}_{\mathsf{lp}}$) and crs (that can depend on lp and thus $\mathcal{L}_{\mathsf{lp}}$), it must be hard to generate $(C, \mathbb{x}, \pi)$, such that the verifier accepts $(C, \mathbb{x}, \pi)$ but $(C, \mathbb{x}) \notin \mathcal{L}_{\mathsf{lp}}$.

DGPRS [18] and FLPS [19] are pairing-based C&P QA-SNARGs for certain constraint systems. DGPRS and FLPS use a perfectly binding commitment scheme Γ and two more building blocks:

(1) a succinct functional somewhere-extractable (FSE, [19]) commitment scheme to commit to $(\mathbb{x}, \mathbb{w})$. FSE satisfies the following property: for a small locality parameter q, one can invisibly reprogram FSE's commitment key FSE.ck

[2] The initial QA-NIZK constructions were for linear subspaces [40,45]. They (and the bilateral linear subspace QA-SNARG, used in [18,19] and the current paper) have a language parameter that is not a commitment key. We use the acronym QA-SNARG since it fits our framework better.

so that one can later black-box "somewhere-extract" the desired q linear combinations of the coefficients of $[\mathbb{x}, \mathbb{w}]_1$.[3]

(2) a succinct bilateral subspace QA-SNARG argument BLS [27] to prove that a tuple of commitments belongs to a specific subspace (e.g., Γ-commitments and FSE-commitments are to the same $(\mathbb{x}, \mathbb{w})$).

DGPRS and FLPS are falsifiable assumption-based, quasi-adaptively sound, for hard languages, and have a succinct argument. This does not contradict Gentry-Wichs since their statement contains a *non-succinct* commitment C from which the reduction can black-box extract the witness. (See Appendix A.1.)

Consider their soundness proof to understand why DGPRS and FLPS are quasi-adaptively sound. Assume that an adversary $\mathcal{A}$ broke the quasi-adaptive soundness by outputting an accepting $(C, \mathbb{x}, \pi)$. Thus, either (1) C is not a commitment to $(\mathbb{x}, \mathbb{w})$ for any $\mathbb{w}$, or (2) at least one constraint is unsatisfied (C commits to $(\mathbb{x}, \mathbb{w})$ for some $\mathbb{w}$, but $\mathbb{w}$ is not a correct witness for $\mathbb{x} \in \mathcal{L}_{\mathsf{lp}}$). DGPRS and FLPS define two reductions $\mathcal{B}_1$ and $\mathcal{B}_2$. $\mathcal{B}_1$ is a reduction to the BLS security, guaranteeing in particular that (1) cannot happen.

Let us focus on $\mathcal{B}_2$. $\mathcal{B}_2$ samples a constraint number $\varrho \leftarrow\!\!\$\ [1, n]$, where n is the number of constraints in the underlying constraint system. $\mathcal{B}_2$ reprograms the CRS to depend on ϱ while $\mathsf{lp} = \Gamma.\mathsf{ck}$ stays unchanged. It follows from the properties of FSE that the CRS hides ϱ. After obtaining $(C, \mathbb{x})$ from $\mathcal{A}$, $\mathcal{B}_2$ black-box extracts from the perfectly binding commitment C all variables, involved in the ϱth constraint. $\mathcal{B}_2$ guesses that the ϱth constraint is unsatisfied and then uses the extracted values to check whether its guess is correct. If the guess is incorrect (i.e., the ϱth constraint is satisfied), then $\mathcal{B}_2$ aborts. Since C is perfectly binding, the adversary's witness $\mathbb{w}$ is fixed by C. Thus, the index of the unsatisfied constraint does not depend on ϱ. (If the adversary can open C to a different message after the CRS reprogramming, one can distinguish the CRSs. The latter is intractable because of the properties of FSE, [18,19].)

Since the index of the unsatisfied constraint does not depend on $\mathcal{B}_2$'s guess ϱ, $\mathcal{B}_2$ aborts with probability $\leq 1 - 1/n$. In the case of non-abortion, $\mathcal{B}_2$ uses FSE's somewhere-extractor to black-box extract a succinct partial witness $[\mathbb{p}^{\varrho}]_1$ from a succinct FSE commitment (also output by $\mathcal{A}$). Here, $[\mathbb{p}^{\varrho}]_1$ is sufficient to verify whether the ϱth constraint of the constraint system is satisfied. The BLS argument (via reduction $\mathcal{B}_1$) guarantees that the values extracted from C are consistent with $[\mathbb{p}^{\varrho}]_1$. $\mathcal{B}_2$ then uses $[\mathbb{p}^{\varrho}]_1$ to break a falsifiable assumption.

FANA. Lipmaa and Pavlyk [52] improve on DGPRS and FLPS in several ways. Their non-C&P zk-SNARG FANA handles the standard R1CS constraint system [25] instead of SSP and SAP used in DGPRS and FLPS, has soundness based on a more plausible falsifiable assumption QALINRES, and is subversion zero-knowledge [6,1,21,2,3] (zero-knowledge even when lp and crs are maliciously constructed). [52] claims that FANA is non-adaptively sound and thus Gentry-Wichs is tight. We only focus on the last claim.

[3] We use the standard additive bracket notation for pairings. For example, for $s \in \mathbb{Z}_p$, $[s]_1 = s[1]_1 \in \mathbb{G}_1$. See Section 3.

FANA omits ck and the commitment C. FANA's security reduction black-box extracts partial witnesses $[\mathbb{p}^{\varrho}]_1$ from the FSE commitment. As in DGPRS and FLPS, extraction is done after reprogramming the CRS. To ensure that $[\mathbb{p}^{\varrho}]_1$ does not covertly depend on ϱ, [52] reverts to non-adaptivity, assuming that the statement $\mathbb{x}$ (recall that there is no commitment C) is fixed before the CRS is created. [52] argues that since $\mathbb{x}$ does not depend on crs, neither does the index ϱ of an unsatisfied constraint; hence, a slight modification of the quasi-adaptive soundness proof of [18,19] goes through.

2.2 FANA Is Not Sound

FANA's soundness proof states that since the statement $\mathbb{x}$ is fixed, the unsatisfied constraint number ϱ does not depend on the CRS. Next, we will explain why one cannot assume that the number ϱ of the (possibly only) unsatisfied constraint did not change after the CRS reprogramming.

If C is a perfectly binding commitment as in DGPRS and FLPS, then one can use the properties of FSE to guarantee that one cannot open C to a different value after the CRS reprogramming. Using a succinct FSE commitment as in FANA, the committed message can change with each CRS reprogramming. So, one cannot ensure that the partial witnesses are consistent. More precisely, one cannot break a falsifiable assumption with a black-box reduction if the partial witnesses are inconsistent (a non-black-box reduction might still be possible). FANA's security proof does not guarantee that the adversary uses the same full witness $\mathbb{w}$ after each reprogramming; in particular, there is no guarantee that ϱ did not change. If ϱ changed, one could not argue that the non-abortion probability in the soundness reduction is at least $1/n$. Indeed, this probability might be zero when the adversary leaves some constraint unsatisfied, but the number of this constraint depends on the CRS in a non-trivial manner.

2.3 Semi-Adaptive Black-Box Knowledge-Soundness

An argument system is *black-box knowledge-sound* if, for every PPT prover, there exists a black-box PPT extractor $\mathsf{Ext}_{\mathsf{ks}}$ such that if the prover convinces the verifier to accept a statement $\mathbb{x}$ with a non-negligible probability, then $\mathsf{Ext}_{\mathsf{ks}}$ extracts a witness $\mathbb{w}$ for the validity of $\mathbb{x}$. In an adaptively sound SNARG, since the prover's message is much shorter than the witness, one cannot black-box extract a witness from a single argument. An alternative approach is to extract a witness directly from the code of the prover. In all existing solutions, such *non-black-box extraction* is enabled by non-falsifiable knowledge assumptions.

One can achieve black-box extractable *interactive* succinct arguments by allowing rewinding the prover to earlier rounds. Rewinding gives the extractor power to run the prover with different verifier's randomnesses ϱ and thus obtain many succinct arguments π^{ϱ}. From π^{ϱ}, the extractor can "somewhere extract" a partial witness $\mathbb{p}^{\varrho}$. If the total length of different arguments is larger than the witness length, one does not have the information-theoretic barrier anymore and can thus potentially compute $\mathbb{w}$ from $\{\mathbb{p}^{\varrho}\}$ and thus black-box extract $\mathbb{w}$.

In the interactive case, one usually splits this procedure into two parts: the rewinding step to obtain many transcripts tr^{ϱ} (that, in particular, contain π^{ϱ}) and the gluing step that inputs the transcripts and outputs the full witness $\mathbb{w}$. One formalizes the second step by defining special soundness [16] and saying that the argument is special-sound if the second step succeeds. The first step essentially reduces knowledge-soundness to special soundness. We use the same two-step methodology, albeit for non-interactive semi-adaptive arguments.

In adaptively sound SNARGs, the prover can be rewound to the point before it creates the argument π. The extractor will not have more power since π is not rerandomized by the verifier. In non-adaptively sound SNARGs, one can rewind to the point before the CRS was created. One can then use a new randomness ϱ to create a new CRS crs^{ϱ} and obtain a new succinct argument π^{ϱ}. From π^{ϱ}, the extractor can "somewhere extract" a partial witness $\mathbb{p}^{\varrho}$. Similarly to the interactive case, one can thus breach the information-theoretic barrier. However, a malicious prover can compute each argument using a different witness; this is one intuition behind the impossibility result of [12] that falsifiable assumption-based non-adaptively knowledge-sound SNARKs for NP are impossible.

Local and global consistency. If the underlying language is a constraint system, one can think of $\varrho := \mathcal{S}$ as a set of constraints and $\mathbb{p}^{\varrho} = \mathbb{p}_{\mathcal{S}}$ a partial witness that satisfies all constraints in the set $\mathcal{S}$. If this holds for every (small) $\mathcal{S}$, the SNARG satisfies *local consistency* [41]. For *global consistency*, one would like the partial witness $\mathbb{p}_{\mathcal{S}}$ to be consistent with some full witness $\mathbb{w}$, $\mathbb{p}_{\mathcal{S}}(\mathcal{S}) = \mathbb{w}|_{\mathcal{S}}$. In particular, all partial witnesses should be mutually consistent. ([41] does not satisfy global consistency.) We will give more details in Section 5.1.

Semi-adaptive black-box knowledge-soundness. Non-adaptively sound SNARKs can be seen as two-message protocols, where the first message is the CRS, and the second message is the argument. A logical approach to overcome their impossibility result while still staying in the realm of black-box extraction is to increase the number of rewinding points (or messages). C&P SNARKs are a natural way of doing that: they can be seen as four-message protocols, where the first message is a commitment key (also known as the language parameter), the second message is a commitment C and a statement $\mathbb{x}$, the third message is a CRS, and the fourth message is an argument. However, the CRS does not depend on the second message; moreover, the same CRS can be used in different SNARKs by different provers. Thus, semi-adaptivity can be seen as a trust assumption that $(C, \mathbb{x})$ does not depend on the CRS. We use the name *semi-adaptive* since the adversary is allowed to output the statement after seeing ck (the first half of the trusted parameters) and before seeing crs (another half). See Fig. 1.

Again, the extractor can repeatedly use a new ϱ to create a new CRS crs^{ϱ} and obtain a succinct argument π^{ϱ}. From π^{ϱ}, the extractor can "somewhere extract" a partial witness $\mathbb{p}^{\varrho}$. In our soundness proof, we do not rewind the creation of ck. The difference with the non-adaptive case is that we have the commitment C that must be the same in different rewindings. So, all partial witnesses must be consistent with C. If they are also consistent with each other, we can compute a

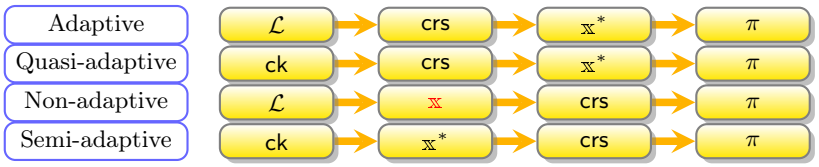


Fig. 1. C&P SNARKs: temporal dependencies. In the case of quasi-adaptive and semi-adaptive soundness, $\mathsf{lp} = \mathsf{ck}$ also fixes the language $\mathcal{L}_{\mathsf{lp}}$. Here, $\mathbb{x}^* = (C, \mathbb{x})$, where C is a commitment. Non-adaptive soundness differs since $\mathbb{x}$ is created before any trusted parameters (ck or crs), which means that $\mathbb{x}$ cannot contain a commitment.

full witness $\mathbb{w}$ that is consistent with all partial witnesses and thus satisfies all constraints. Semi-adaptive knowledge-soundness states that this must always be possible. In the soundness proof of the new SNARK, we construct a reduction that works if this is false (i.e., two partial witnesses are not mutually consistent).

Definition. We define a new security notion for C&P SNARKs, *semi-adaptive black-box G-knowledge-soundness* that insists that one can efficiently construct $G(\mathbb{w})$ given oracle access to the prover that outputs arguments corresponding to the fixed ck, C, $\mathbb{x}$ but different CRSs. Its definition is inspired by non-adaptive black-box knowledge-soundness in [12] and witness-extended emulation (WEE, [48]). In particular, if an adversary outputs a single accepting transcript, the black-box extractor outputs both the accepting transcript (from the correct distribution) and $G(\mathbb{w})$ with a similar probability. Here, G is a permutation that plays a similar role to G in Groth-Sahai proofs [33] (that are usually G-extractable) and G-unforgeable signature schemes [5]. In our new SNARK for R1CS, $G(s) := [sy]_1$ for a trapdoor y. When handling SSP [17] (Boolean circuits) instead of R1CS, one can set $G(s) = s$.

Applications of semi-adaptivity. As argued above, black-box G-knowledge-soundness is a natural security notion that seems to be the best one can do in the context of SNARKs, given the impossibility results of [26,12]. It is a semi-adaptive version of the non-adaptive black-box knowledge-soundness of [12].

Semi-adaptive knowledge-sound SNARKs have natural applications. Consider, for example, e-voting for national institutions like the parliament, where the (universal and updatable) commitment key is made public before elections. The commitment key might be used in other applications and thus has to be created highly securely. In a concrete election, the voters can first commit to their ballot, the trusted third parties can create a non-universal CRS (that may depend on the ballot structure and say the number of voters), and then each voter can construct an argument, proving that the ballot is correct. When using our results, the SNARK relies on falsifiable assumptions. Using weak assumptions is vital for national security. Proving all NP statements is essential in the case of complex ballot structures. Practical efficiency, as provided by Punic, is essential for the SNARK to be used at all. Creating the CRS after the commitment phase seems a natural compromise to achieve all the other properties.

CRS-Indistinguishability. To prove black-box knowledge-soundness, we need that any adversary that makes the verifier accept with a non-negligible probability must succeed with non-negligible probability for *every* argument ϱ to $\mathsf{K}_{\mathsf{crs}}$. Only then will $\mathsf{Ext}_{\mathsf{ks}}$ be able to retrieve all partial witnesses needed to output $G(\mathbb{w})$. To tackle this, it suffices to assume that the CRSs, corresponding to any two values of ϱ, are indistinguishable.

Special Soundness. We define semi-adaptive computational (k, G)-special soundness, stating that there exists a black-box PPT extractor $\mathsf{Ext}_{\mathsf{ss}}$, such that if an adversary outputs k consistent transcripts $\mathsf{tr}^{\varrho} = (C, \mathbb{x}, \mathsf{crs}^{\varrho}, \mathsf{td}^{\varrho}, \pi^{\varrho})$ with pairwise distinct ϱ, then $\mathsf{Ext}_{\mathsf{ss}}$ outputs $G(\mathbb{w})$. We prove that *any* semi-adaptively computationally (k, G)-special-sound and CRS-indistinguishable C&P zk-SNARK is semi-adaptively black-box G-knowledge-sound. Thus, it suffices to prove that a zk-SNARK satisfies the first two properties.

2.4 New SNARK

Construction. Since FANA only uses FSE commitments (with commitment keys reprogrammed by the reduction), it is not semi-adaptively sound. We construct Punic, a falsifiable assumption-based semi-adaptively sound C&P SNARK with a succinct commitment. Punic is CRS-indistinguishable and semi-adaptively black-box G-knowledge-sound for $G(s) := [sy]_1$, where y is a trapdoor. G involves scalar multiplication since the extractor retrieves a group element and the DL is hard; we need y due to using FSE and VCF. Moreover, G is needed since we deal with R1CS (i.e., arithmetic circuits). As we note in Section 2.6, in the case of SSP [17] (Boolean circuits), G can be the identity map.

Punic is a variant of FANA, to which we add a language parameter lp (vector commitment scheme's commitment key) and a vector commitment $[C]_1$ to the witness. Alternatively, Punic is an (optimized) variant of FLPS that replaces the perfectly-binding commitment scheme with a well-chosen *vector* commitment scheme VCF. Our completeness, zero-knowledge, and CRS-indistinguishability proofs are relatively straightforward. We will next explain the soundness proof.

Soundness Proof. Recall that it suffices to prove special soundness. In the special soundness proof, we fix $\mathsf{lp} = \mathrm{VCF.ck}$, where VCF is a new vector commitment scheme, described later. In [18,19], one fixes the adversary's statement (a vector commitment $[C]_1$ to $(\mathbb{x}, \mathbb{w})$, and an R1CS statement $\mathbb{x}$). Then, the reduction $\mathcal{B}$ samples $\varrho \leftarrow\!\!\$\ [1, n]$, reprograms the CRS accordingly, runs the soundness adversary $\mathcal{A}$ once, and guesses the ϱth constraint is violated. If the guess is wrong, $\mathcal{B}$ aborts. This guarantees local consistency (for every ϱ, a partial witness exists that satisfies the ϱth constraint). [18,19] guarantee soundness (the existence of a full witness $\mathbb{w}$, consistent with each partial witness) by using a perfectly binding commitment to $(\mathbb{x}, \mathbb{w})$ and checking its consistency with partial witnesses.

We use a different proof strategy since we do not have a perfectly binding commitment. Our special soundness reduction $\mathcal{B}$ inputs n transcripts tr^{ϱ}. For each ϱ, $\mathcal{B}$ uses FSE to black-box extract a partial witness $G(\mathbb{p}^{\varrho})$ allowing to check

whether the ϱth constraint is satisfied. For this, $\mathcal{B}$ reprograms FSE's commitment key, which is part of Punic's CRS. The verification equation ascertains that $G(\mathbb{p}^\varrho)$ is consistent with the value committed to by $[C]_1$.

More precisely, we construct a special soundness extractor $\mathsf{Ext}_{\mathsf{ss}}$ that computes $G(\mathbb{w})$ given partial witnesses $G(\mathbb{p}^\varrho)$ output by the FSE black-box somewhere extractor. When $\mathsf{Ext}_{\mathsf{ss}}$ fails, we construct three reductions, two of which are inspired by the reductions in [18,19,52] (we briefly described them above). The third reduction works when for each ϱ, $\mathbb{p}^\varrho$ satisfies the ϱth constraint, but $\mathsf{Ext}_{\mathsf{ss}}$ fails to output $G(\mathbb{w})$ where $\mathbb{w}$ satisfies all constraints. Then, at least two partial witnesses (say, $\mathbb{p}^i$ and $\mathbb{p}^j$) must be inconsistent.

The crux of our solution is using FSE to black-box extract well-defined information, allowing us to build a reduction out of this inconsistency. Let $N(\varrho)$ be the set of witness coefficients used in the ϱth constraint. For all $k \in N(\varrho)$, we use FSE to black-box extract VCF's local opening and local proof for the kth coefficient of the full witness. We need a vector commitment scheme precisely for the existence of local proofs. Since we black-box extract both local openings and local proofs by using FSE, VCF needs to satisfy two novel requirements:

(a) *full algebraicity:* one can compute the vector commitment, the local opening (the claimed vector coefficient), and the local proof from $(\mathbb{x}, \mathbb{w})$ and the commitment randomizer by using linear maps,
(b) *F-position-binding:* position-binding even for an adversary who, instead of coordinates $\eta \neq \eta'$, outputs $F(\eta) \neq F(\eta')$, for a permutation F. We need it since FSE is F-extractable, allowing one to extract only $F(\eta) := [\eta]_1$.

In the ϱth loop of the reduction, we reprogram the CRS so that we can black-box extract $(G(\eta_k^\varrho), [\varphi_k^\varrho]_1)$ for $k \in N(\varrho)$. Here, $G(\eta_k^\varrho) = [\eta_k^\varrho y]_1$ and $[\varphi_k^\varrho]_1$ are the local opening and the local proof of the full witness $\mathbb{w}^\varrho$ the adversary used in the ϱth iteration. If $\mathbb{p}^i \neq \mathbb{p}^j$ for some i, j, then we extract two openings $(G(\eta_k^i), [\varphi_k^i]_1)$ and $(G(\eta_k^j), [\varphi_k^j]_1)$, such that $\eta_k^i \neq \eta_k^j$, breaking G/F-position-binding. Assuming F-position-binding, all extracted partial witnesses are consistent. Using a greedy algorithm, we efficiently compute $G(\mathbb{w})$ from $\{G(\mathbb{p}^\varrho)\}$. QED.

This is a novel proof technique for handling the case when partial witnesses exist. We hope it will find other applications like in SNARGs for P or batch arguments for NP [41,29,15]. A drawback is that we must extract all coefficients at a constraint, so each R1CS constraint must have a small locality. Any R1CS instance can be modified to be such by introducing new constraints using standard techniques. Such a restriction is well-known and used in several efficient zk-SNARKs, [23,57]. [41] used 3CNF (with locality three) for a similar reason.

Punic's black-box G-knowledge-soundness relies on several falsifiable bilinear group assumptions, of which QALINRES [52] is the most complicated. As proven in [52], QALINRES is secure in the algebraic group model (AGM [22]); for completeness, we reprove this result.

On No-Signaling. Obtaining global consistency *efficiently* from local consistency is a major open problem in constructing falsifiable assumption-based SNARGs. One approach [42,41,29,15] is to use no-signaling PCPs and commitments. However, this approach usually works only for memory-bound computations; one

has to use additional techniques in the general case. Our approach to achieving global consistency has direct advantages compared to no-signaling commitments. See Appendix A.2 for a discussion.

2.5 Fully Algebraic F-Position-Binding Vector Commitment

Punic uses a vector commitment scheme VCF. To use FSE to black-box extract VCF's local openings and local proofs, VCF must be fully algebraic and F-position-binding. Both properties seem novel for vector commitment schemes, though they are similar to known requirements on other primitives (e.g., algebraic commitments and F-unforgeable signature schemes [5]).

VCF is based on the CDHK vector commitment scheme [10]. We show CDHK is fully-algebraic but not F-position-binding. We introduce a new trapdoor y (explaining the choice of G) and a knowledge component without making VCF less efficient. VCF remains fully algebraic. We prove VCF is F-position-binding under a new but standard-looking assumption VCSDH (*Vector Commitment Strong Diffie-Hellman*). We reduce VCSDH to QALINRES.

We hope the new notion of fully-algebraic and/or F-position-binding vector commitments will have independent applications.

2.6 Efficiency

We explicitly strived to make Punic concretely efficient. Its prover computation is dominated by $\Theta(n)$ group operations, and the argument size and verifier computation are $\Theta_\lambda(1)$ with small constants. Notably, using vector commitments allows us (differently from [58,12]) to avoid heavy machinery like FHE, hash trees, iO, PCP, and SNARK recursion. In our application, efficiency is difficult to achieve: having larger argument sizes, one can black-box extract more information at a time, making achieving global consistency less difficult. (See comparison with no-signaling commitments in Appendix A.2.)

With some loss in efficiency, one can construct a semi-adaptively sound SNARK based on weaker assumptions. One can use (1) better-known somewhere-extractable commitments [34] known to exist under various assumptions instead of FSE commitments and (2) hash trees instead of the new vector commitment scheme. On the other hand, we do not know how to instantiate linear subspace arguments efficiently on general assumptions.

Kilian. In Appendix A.3, we discuss a solution based on Kilian's seminal interactive zero-knowledge argument. We will leave generalizations for future work.

3 Preliminaries

Let p be a large prime. Denote $\mathbb{F} := \mathbb{Z}_p$. For $\boldsymbol{a} \in \mathbb{F}^m$ and $\mathcal{S} \subseteq [1, m]$, let $\boldsymbol{a}|_{\mathcal{S}} := (a_i)_{i \in \mathcal{S}}$. For two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, let $\boldsymbol{a} \circ \boldsymbol{b}$ be their Hadamard product,

with $(\boldsymbol{a} \circ \boldsymbol{b})_i = a_i b_i$. For a matrix $\boldsymbol{A} = (A_{ij})$, $\boldsymbol{A}_i$ denotes its ith row and $\boldsymbol{A}^{(j)}$ denotes its jth column. Let colspace($\boldsymbol{A}$) be the column space of $\boldsymbol{A}$.

PPT denotes probabilistic polynomial-time; $\lambda \in \mathbb{N}$ is the security parameter. We assume all adversaries are stateful, i.e., keep up a state between different executions. For an algorithm $\mathcal{A}$, range($\mathcal{A}$) is the range of $\mathcal{A}$, i.e., the set of valid outputs of $\mathcal{A}$, $\mathsf{RND}_\lambda(\mathcal{A})$ denotes the random tape of $\mathcal{A}$ (for given λ), and $r \leftarrow\!\!\$\ \mathsf{RND}_\lambda(\mathcal{A})$ denotes the uniformly random choice of r from $\mathsf{RND}_\lambda(\mathcal{A})$. By $s \leftarrow \mathcal{A}(\mathbb{x}; r)$ we denote the fact that $\mathcal{A}$, given an input $\mathbb{x}$ and a randomizer r, outputs s. Let $\mathsf{negl}(\lambda)$ be an arbitrary negligible function, and $\mathsf{poly}(\lambda)$ be an arbitrary polynomial function. We write $a \approx_\lambda b$ if $|a - b| \le \mathsf{negl}(\lambda)$.

Assume $n \mid (p-1)$ is a power of two. Let ω be the nth primitive root of unity modulo p and let $\mathbb{H} := \langle \omega \rangle = \{\omega^{i-1}\}_{i=1}^n$ be a subgroup of $\mathbb{F}^*$. Let $Z_{\mathbb{H}}(X) := \prod_{i=1}^n (X - \omega^{i-1}) = X^n - 1$ be the unique degree n monic polynomial, such that $Z_{\mathbb{H}}(\omega^{i-1}) = 0$ for all $i \in [1, n]$. For $i \in [1, n]$, let $\ell_i(X)$ be the *ith Lagrange polynomial*, that is, the unique degree-$(n-1)$ polynomial, such that $\ell_i(\omega^{i-1}) = 1$ and $\ell_i(\omega^{j-1}) = 0$ for $i \neq j$. Then, $\ell_i(X) = (X^n - 1)\omega^{i-1} / \left(n(X - \omega^{i-1})\right)$.

Cryptography. A *bilinear group generator* $\mathsf{Pgen}(1^\lambda)$ returns $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, [1]_1, [1]_2)$, where $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ are additive cyclic (thus, abelian) groups of prime order p, $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficient non-degenerate bilinear pairing, and $[1]_\gamma$ is a fixed generator of $\mathbb{G}_\gamma$. While $[1]_\gamma$ is a part of p, for the sake of clarity, we often give it as an explicit input to different algorithms. We assume $n \mid (p-1)$, where n is a large deterministically fixed upper bound on the size of the statements that one handles in this bilinear group. The bilinear pairing is of Type-3; that is, there is no efficient isomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$. We use the standard bracket notation: for $\gamma \in \{1, 2, T\}$, we write $[a]_\gamma$ to denote $a[1]_\gamma$. We denote $\hat{e}([a]_1, [b]_2)$ by $[a]_1 \bullet [b]_2$. We mix freely bracket and matrix notation, e.g., $\boldsymbol{AB} = \boldsymbol{C}$ iff $[\boldsymbol{A}]_1 \bullet [\boldsymbol{B}]_2 = [\boldsymbol{C}]_T$. We denote $[\boldsymbol{A}]_2 \bullet [\boldsymbol{B}]_1 := [\boldsymbol{AB}]_T = ([\boldsymbol{B}]_1^\intercal \bullet [\boldsymbol{A}]_2^\intercal)^\intercal$.

Let $\gamma \in \{1, 2\}$. *$DDH_{\mathbb{G}_\gamma}$ (Decisional Diffie-Hellman)* holds relative to Pgen, if for all PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{ddh}}_{\mathsf{Pgen}, \mathbb{G}_\gamma, \mathcal{A}}(\lambda) :=$

$$\Pr\left[\mathcal{A}(\mathsf{p}, [x, y, xy + \beta z]_\gamma) = \beta \mid \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); x, y, z \leftarrow\!\!\$\ \mathbb{F}; \beta \leftarrow\!\!\$\ \{0, 1\}\right] \approx_\lambda \tfrac{1}{2}\ .$$

Let $\kappa^*, \kappa \in \mathbb{N}_+$, with $\kappa^* \ge \kappa$, be small constants. A PPT-sampleable distribution $\mathcal{D}_{\kappa^*, \kappa}$ is a *matrix distribution* if it samples matrices $\boldsymbol{A} \in \mathbb{F}^{\kappa^* \times \kappa}$ of full rank κ. $\mathcal{D}_{\kappa^*, \kappa}$ is *robust* [40] if it samples matrices $\boldsymbol{A}$ whose upper $\kappa \times \kappa$ submatrix $\bar{\boldsymbol{A}}$ is invertible. Denote the lower $(\kappa^* - \kappa) \times \kappa$ submatrix of $\boldsymbol{A}$ by $\underline{\boldsymbol{A}}$. Let $\mathcal{D}_\kappa := \mathcal{D}_{\kappa+1, \kappa}$. $\mathcal{D}_{\kappa^*, \kappa}$-SKerMDH (Split Kernel Diffie-Hellman, [27]) holds relative to Pgen, if for all PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{skermdh}}_{\mathsf{Pgen}, \mathbb{G}_\gamma, \mathcal{D}_{\kappa^*, \kappa}, \mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|l} \boldsymbol{A}^\intercal(\boldsymbol{x}_1 - \boldsymbol{x}_2) = \boldsymbol{0}_\kappa \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \boldsymbol{A} \leftarrow\!\!\$\ \mathcal{D}_{\kappa^*, \kappa}; \\ \boldsymbol{x}_1 - \boldsymbol{x}_2 \neq \boldsymbol{0}_{\kappa^*} & ([\boldsymbol{x}_1]_1, [\boldsymbol{x}_2]_2) \leftarrow \mathcal{A}(\mathsf{p}, [\boldsymbol{A}]_1, [\boldsymbol{A}]_2) \end{array}\right] \approx_\lambda 0\ .$$

The QALINRES Assumption . The new zk-SNARK relies on the *n-Quadratic Arithmetic Linear Residuosity (n-*QALINRES*)* assumption from [52].

Definition 1. n-Quadratic Arithmetic Linear Residuosity (n-QALINRES, [52]) *holds relative to* Pgen*, if for all PPT* $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{qalinres}}_{\mathsf{Pgen},n,\mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|l} \pi = \left(j, [\mathsf{a}, \hat{\eta}_\mathsf{a}, \varphi_\mathsf{a}, \mathsf{c}, \hat{\eta}_\mathsf{c}, \varphi_\mathsf{c}, \mathsf{h}]_1, [\mathsf{b}, \hat{\eta}_\mathsf{b}, \varphi_\mathsf{b}]_2\right) \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \\ \mathsf{a} = \varphi_\mathsf{a}(x - \omega^{j-1}) + \hat{\eta}_\mathsf{a}/y \wedge & x \leftarrow\!\!\$\ \mathbb{F} \setminus \mathbb{H}; y \leftarrow\!\!\$\ \mathbb{F}^*; \\ \mathsf{b} = \varphi_\mathsf{b}(x - \omega^{j-1}) + \hat{\eta}_\mathsf{b}/y \wedge & \mathsf{ck} \leftarrow ([(x^i)_{i=0}^n, y]_\gamma)_{\gamma=1}^2; \\ \mathsf{c} = \varphi_\mathsf{c}(x - \omega^{j-1}) + \hat{\eta}_\mathsf{c}/y \wedge & \pi \leftarrow \mathcal{A}(\mathsf{ck}) \\ \mathsf{ab} - \mathsf{c} = \mathsf{h}Z_\mathbb{H}(x) \wedge \hat{\eta}_\mathsf{a}\hat{\eta}_\mathsf{b} \neq \hat{\eta}_\mathsf{c} y & \end{array}\right] \approx_\lambda 0 \ .$$

QALINRES was introduced in [52] as a more realistic version of TSDH-like assumptions used in [18,19]. In particular, it does not rely on $\mathcal{A}$ outputting elements of $\mathbb{G}_T$. See [52] and Appendix B.2 for a discussion. QALINRES is not publicly verifiable, but it has an efficient challenger.

Lipmaa and Pavlyk [52] proved that QALINRES is secure in the AGM under the PDL assumption. Since [52] does not include this proof, we reprove it in Appendix C. We stress that while the AGM is an idealized model that can be used to prove non-falsifiable assumptions, QALINRES itself is a falsifiable assumption. QALINRES is non-interactive. Moreover, QALINRES is a "Maurer-game" [60], and thus the specific AGM criticisms of [60,61] do not apply to it.

3.1 Underlying Commitment Schemes

We use several commitment schemes. Each commitment scheme has PPT algorithms $\mathsf{Pgen} : 1^\lambda \mapsto \mathsf{p}$ (for parameter generation), $\mathsf{K}_\mathsf{ck} : (\mathsf{p}, n) \mapsto (\mathsf{ck}, \mathsf{td})$ (for key generation; here, n is the vector length) and $\mathsf{Com} : (\mathsf{ck}, \boldsymbol{\mu}; r) \mapsto (C, D)$ (for commitment; D is the decommitment information). Let $\mathcal{M}$ be the message space, $\mathcal{C}$ the commitment space, and $\mathcal{R}$ the randomizer space. To simplify notation, we always assume ck implicitly contains p.

Vector Commitment. Let $\mathcal{D}$ be a domain. A vector commitment scheme $\Gamma = (\mathsf{Pgen}, \mathsf{K}_\mathsf{ck}, \mathsf{Com}, \mathsf{LOpen}, \mathsf{LVer})$ is a commitment scheme, with $\mathcal{M} = \mathcal{D}^n$ for $n \leq \mathsf{poly}(\lambda)$, that has two additional algorithms [47,36,13]:

Local opening: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathsf{ck} \in \mathsf{K}_\mathsf{ck}(\mathsf{p}, n)$, commitment $C \in \mathcal{C}$, index $j \in [1, n]$, and decommitment information D, $\mathsf{LOpen}(\mathsf{ck}, C, j, D)$ returns (η, φ), where η (local opening) is a candidate for μ_j and φ is a local proof.

Local verification: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathsf{ck} \in \mathsf{K}_\mathsf{ck}(\mathsf{p}, n)$, commitment $C \in \mathcal{C}$, index $j \in [1, n]$, candidate value η for μ_j, and local proof φ, $\mathsf{LVer}(\mathsf{ck}, C, j, \eta, \varphi)$ returns either 0 or 1.

Γ must be complete according to the natural definition ($\mathsf{LVer}(\mathsf{p}, \mathsf{ck}, C, j, \eta, \varphi) = 1$ for $(\eta, \varphi) \leftarrow \mathsf{LOpen}(\mathsf{ck}, C, j, D)$ and $(C, D) \leftarrow \mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r)$). Γ must satisfy the following security properties.

Position-binding: for all λ, PPT $\mathcal{A}$, and $n \in \mathsf{poly}(\lambda)$, $\mathsf{Adv}^{\mathrm{posb}}_{\mathsf{Pgen},n,\Gamma,\mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|l} \eta_0 \neq \eta_1 \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \\ \mathsf{LVer}(\mathsf{ck}, C, j, \eta_0, \varphi_0) = 1 \wedge & (\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{K}_\mathsf{ck}(\mathsf{p}, n); \\ \mathsf{LVer}(\mathsf{ck}, C, j, \eta_1, \varphi_1) = 1 & (C, j, \eta_0, \eta_1, \varphi_0, \varphi_1) \leftarrow \mathcal{A}(\mathsf{ck}) \end{array}\right] \approx_\lambda 0 \ .$$

$\mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n)$: $x \leftarrow\!\!\$\ \mathbb{F} \setminus \mathbb{H}$; $\mathsf{td} \leftarrow x$; $\mathsf{ck} \leftarrow ([(x^i)_{i=0}^n]_\gamma, [1, x]_{3-\gamma})$;
store $[\mathsf{ck}_\ell]_\gamma \leftarrow [\ell_1(x), \ldots, \ell_n(x), Z_{\mathbb{H}}(x)]_\gamma$; return $(\mathsf{ck}, \mathsf{td})$;
$\mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r)$: $r \leftarrow\!\!\$\ \mathbb{F}$; $[C(x)]_\gamma \leftarrow [\mathsf{ck}_\ell]_\gamma \cdot \binom{\boldsymbol{\mu}}{r} = \sum_{i=1}^n \mu_i [\ell_i(x)]_\gamma + r[Z_{\mathbb{H}}(x)]_\gamma$;
return $([C(x)]_\gamma, (\boldsymbol{\mu}, r))$; // (C, D)
$\mathsf{LOpen}(\mathsf{ck}, [C(x)]_\gamma, j, (\boldsymbol{\mu}, r))$: $\eta \leftarrow \mu_j$; $[\varphi(x)]_\gamma \leftarrow [(C(x) - \eta)/(x - \omega^{j-1})]_\gamma$;
return $(\eta, [\varphi(x)]_\gamma)$;
$\mathsf{LVer}(\mathsf{ck}, [C(x)]_\gamma, j, \eta, [\varphi(x)]_\gamma)$:
check that $[C(x) - \eta]_\gamma \bullet [1]_{3-\gamma} = [\varphi(x)]_\gamma \bullet [x - \omega^{j-1}]_{3-\gamma}$;
$\mathsf{Sim}(\mathsf{ck}, \mathsf{td} = x, \{j_i\}_{i \in I}, \{\mu_{j_i}\}_{i \in I})$: $r \leftarrow\!\!\$\ \mathbb{F}$; $r' \leftarrow (\sum_{i \in I} \mu_{j_i} \ell_{j_i}(x))/Z_{\mathbb{H}}(x) + r$;
$[C(x)]_\gamma \leftarrow \mathsf{Com}(\mathsf{ck}, \mathbf{0}; r') = r'[Z_{\mathbb{H}}(x)]_\gamma$; return $[C(x)]_\gamma$;

Fig. 2. The position-binding vector commitment scheme CDHK.

Perfect zero-knowledge: there exists a PPT simulator Sim, such that for all λ, all $\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda)$, all $(\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n)$, all $\boldsymbol{\mu} \in \mathcal{D}^n$, and any poly-size set $\{j_i \in [1, n]\}_i$, the distributions δ_0 and δ_1 are identical, where

$$\delta_0 := \{(\mathsf{ck}, C, \{\mathsf{LOpen}(\mathsf{ck}, C, j_i, D)\}) : r \leftarrow\!\!\$\ \mathsf{RND}_\lambda(\mathsf{Com}); (C, D) \leftarrow \mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r)\} \ ,$$
$$\delta_1 := \{(\mathsf{ck}, \mathsf{Sim}(\mathsf{ck}, \mathsf{td}, \{j_i\}, \{\mu_{j_i}\}))\} \ .$$

Modeled after the seminal KZG polynomial commitment scheme [43], Camenisch et al. [10] proposed a vector commitment scheme. Let $\mathcal{D} = \mathbb{F}$, $\mathcal{M} = \mathcal{D}^n$, $\mathcal{C} = \mathbb{G}_\gamma$ for $\gamma \in \{1, 2\}$, and $\mathcal{R} = \mathbb{F}$. In Fig. 2, we depict a simplified version CDHK of their scheme. CDHK is position-binding under the standard n-SDH assumption [9]. Straightforwardly, CDHK satisfies perfect zero-knowledge.

FSE *Commitment.* Let $F : \mathcal{M} \to \mathcal{C}$ be a (one-way, p-dependent) permutation. Let $\mathcal{F}$ be a function family, where $f \in \mathcal{F}$ inputs a vector $\boldsymbol{\mu}$ and outputs an element of $\mathcal{C}$. A *functional*[4] *somewhere F-extractable (F-FSE) commitment scheme* [19] $\Gamma = (\mathsf{Pgen}, \mathsf{K}_{\mathsf{ck}}, \mathsf{Com}, \mathsf{swExt})$ for $\mathcal{F}$ allows one to commit to a vector $\boldsymbol{\mu}$, s.t. for any $q \le n$, (1) the commitment key ck depends on q and a function tuple $f_1, \ldots, f_q \in \mathcal{F}$, (2) commitment keys corresponding to different function tuples are computationally indistinguishable, and (3) given the extraction key, one can extract from the commitment the vector $(F(f_1(\boldsymbol{\mu})), \ldots, F(f_q(\boldsymbol{\mu})))$.

More precisely, an F-FSE *commitment scheme* $\Gamma = (\mathsf{Pgen}, \mathsf{K}_{\mathsf{ck}}, \mathsf{Com}, \mathsf{swExt})$ for a function family $\mathcal{F}$ consists of the following (P)PT algorithms.

Parameter generation: $\mathsf{Pgen}(1^\lambda)$ returns p (e.g., the group description).

Commitment key generation: for parameters p, a positive integer $n \le \mathsf{poly}(\lambda)$, a locality parameter $q \in [1, n]$, and a tuple $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$

[4] Defined as functional somewhere statistically binding (SSB) commitment in [19]; generalizes SSB hashes [34,54]. In SSB hashes, $\mathcal{F}$ is the family of point functions, and q is always equal to one. On the other hand, we do not need the local opening property, thus obtaining better efficiency. Since extractability is essential, we call them functional SE. DGPRS and FLPS predate [15]. SE commitments have been used to build SNARGs for P and batch-arguments for NP [29,15].

with $|\mathcal{S}| \le q$, $\mathsf{K_{ck}}(\mathsf{p}, n, q, \mathcal{S})$ outputs a commitment key ck and an extraction key $\mathsf{td} = \mathsf{ek}$. We assume ck and ek implicitly specify p.

Commitment: for a commitment key ck, a message $\boldsymbol{\mu} \in \mathcal{M}^n$, and a randomizer $r \in \mathcal{R}$, $\mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r)$ outputs a commitment $C \in \mathcal{C}$.

Somewhere (black-box) extraction: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, a positive integer $n \le \mathsf{poly}(\lambda)$, a locality parameter $q \in [1, n]$, a tuple $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ with $1 \le |\mathcal{S}| \le q$, $(\mathsf{ck}, \mathsf{ek}) \in \mathsf{K_{ck}}(\mathsf{p}, n, q, \mathcal{S})$, and $C \in \mathcal{C}$, $\mathsf{swExt}(\mathsf{ek}, C)$ returns a tuple $\big(F(f_1(\boldsymbol{\mu})), \ldots, F(f_{|\mathcal{S}|}(\boldsymbol{\mu}))\big) \in \mathcal{M}^{|\mathcal{S}|}$.

For $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ and a vector $\boldsymbol{\mu}$, denote $f_{\mathcal{S}}(\boldsymbol{\mu}) = (f_1(\boldsymbol{\mu}), \ldots, f_{|\mathcal{S}|}(\boldsymbol{\mu}))$.

An F-FSE *commitment scheme* Γ for the function family $\mathcal{F}$ can satisfy the following security requirements.

Function-Set Hiding: for all λ, PPT $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, and $q \in [1, n]$, $\mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{Pgen},\Gamma,n,q,\mathcal{A}}(\lambda) := 2 \cdot |\varepsilon^{\mathsf{fsh}} - 1/2| \approx_\lambda 0$, where $\varepsilon^{\mathsf{fsh}} :=$

$$\Pr\left[\begin{array}{l|l} \beta' = \beta \wedge \mathcal{S}_0, \mathcal{S}_1 \subseteq \mathcal{F} & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); (\mathcal{S}_0, \mathcal{S}_1) \leftarrow \mathcal{A}(\mathsf{p}, n, q); \beta \leftarrow\!\$\ \{0,1\}; \\ \wedge |\mathcal{S}_0|, |\mathcal{S}_1| \le q & (\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{K_{ck}}(\mathsf{p}, n, q, \mathcal{S}_\beta); \beta' \leftarrow \mathcal{A}(\mathsf{ck}) \end{array}\right] .$$

Intuitively, ck reveals computationally no information about $\mathcal{S}$.

Almost Everywhere Perfectly Hiding (AEPH): for all λ, unbounded $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, and $q \in [1, n]$, $\mathsf{Adv}^{\mathrm{aeph}}_{\Gamma,n,q,\mathcal{A}}(\lambda) := 2 \cdot |\varepsilon^{\mathsf{aeph}} - 1/2| = 0$, where $\varepsilon^{\mathsf{aeph}} :=$

$$\Pr\left[\begin{array}{l|l} \beta' = \beta \wedge \mathcal{S} \subseteq \mathcal{F} & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q); \\ \wedge |\mathcal{S}| \le q \wedge & (\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{K_{ck}}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{\mu}_0, \boldsymbol{\mu}_1) \leftarrow \mathcal{A}(\mathsf{ck}); \\ f_{\mathcal{S}}(\boldsymbol{\mu}_0) = f_{\mathcal{S}}(\boldsymbol{\mu}_1) & \beta \leftarrow\!\$\ \{0,1\}; r \leftarrow\!\$\ \mathcal{R}; (C, D) \leftarrow \mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}_\beta; r); \\ & \beta' \leftarrow \mathcal{A}(C) \end{array}\right] .$$

Intuitively, given ck, that depends on $\mathcal{S}$, the commitment hides perfectly the values of μ_j for $j \notin \mathcal{S}$.

Somewhere F-Extractability: for all λ, $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $n \in \mathsf{poly}(\lambda)$, $q \in [1, n]$, $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ with $|\mathcal{S}| \le q$, $(\mathsf{ck}, \mathsf{ek}) \leftarrow \mathsf{K_{ck}}(\mathsf{p}, n, q, \mathcal{S})$, and PPT $\mathcal{A}$,

$$\Pr\left[\mathsf{swExt}(\mathsf{ek}, C) \neq \big(F(f_1(\boldsymbol{\mu})), \ldots, F(f_{|\mathcal{S}|}(\boldsymbol{\mu}))\big) \,\middle|\, \begin{array}{l} (\boldsymbol{\mu}, r) \leftarrow \mathcal{A}(\mathsf{ck}); \\ (C, D) \leftarrow \mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r) \end{array}\right] = 0 .$$

I.e., given ck, that depends on $\mathcal{S}$, and an extraction key, one can black-box extract $F(f_{\mathcal{S}}(\boldsymbol{\mu}))$. ([19] called this property *somewhere perfect F-extractability.*)

Construction. Fix $\gamma \in \{1, 2\}$. Let $F : a \mapsto [a]_\gamma$. In Fig. 3, we depict the FSE scheme of [19] for the family of all linear maps. It represents q linear maps by $[\boldsymbol{M}]_\gamma \in \mathbb{G}_\gamma^{q \times n}$, where each row contains coefficients of one map. Clearly, $[\mathsf{c}]_\gamma \leftarrow \mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r)$ is equal to $\mathsf{ck}\binom{\boldsymbol{\mu}}{r} = \boldsymbol{R}[\boldsymbol{M}']_\gamma\binom{\boldsymbol{\mu}}{r} = \begin{bmatrix} \boldsymbol{R}\boldsymbol{M}\boldsymbol{\mu} \\ \boldsymbol{R}(\boldsymbol{\varrho}^\intercal \boldsymbol{\mu} + r) \end{bmatrix}_\gamma$.

Fact 1 ([19]) *Fix $q < n = \mathsf{poly}(\lambda)$. The scheme in Fig. 3 is (i) function-set hiding relative to Pgen under $DDH_{\mathbb{G}_\gamma}$: for each PPT $\mathcal{A}$, there exists a PPT $\mathcal{B}$, such that $\mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{Pgen},\Gamma,n,q,\mathcal{A}}(\lambda) \le \lceil \log_2(q+1) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathbb{G}_\gamma,\mathsf{Pgen},\mathcal{B}}(\lambda)$. (ii) almost everywhere perfectly-hiding, (iii) somewhere F-extractable for $F = [\cdot]_\gamma$.*

$\mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n, q, [\boldsymbol{M}]_\gamma \in \mathbb{G}_\gamma^{q\times n})$: // $\mathcal{M} = \mathcal{R} = \mathbb{F}^n$ and $\mathcal{C} = \mathbb{G}_\gamma^{q+1}$
sample a full-rank $\boldsymbol{R} \leftarrow\!\$\ \mathbb{F}^{(q+1)\times(q+1)}$; $\boldsymbol{\varrho} \leftarrow\!\$\ \mathbb{F}^n$;
$[\boldsymbol{M}']_\gamma \leftarrow \left[\begin{smallmatrix} \boldsymbol{M} & \mathbf{0} \\ \boldsymbol{\varrho}^\intercal & 1 \end{smallmatrix}\right]_\gamma \in \mathbb{F}^{(q+1)\times(n+1)}$; $\mathsf{ck} \leftarrow \boldsymbol{R}[\boldsymbol{M}']_\gamma \in \mathbb{G}_\gamma^{(q+1)\times(n+1)}$;
$\mathsf{td} = \mathsf{ek} \leftarrow \boldsymbol{R}^{-1}$; **return** $(\mathsf{ck}, \mathsf{td})$;
$\mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu} \in \mathbb{F}^n; r \in \mathbb{F})$: **return** $\mathsf{ck} \cdot \left(\begin{smallmatrix} \boldsymbol{\mu} \\ r \end{smallmatrix}\right)$;
$\mathsf{swExt}(\mathsf{ek}, [\mathsf{c}]_\gamma)$: compute $\left[\begin{smallmatrix} \boldsymbol{M}\boldsymbol{\mu} \\ \boldsymbol{\varrho}^\intercal\boldsymbol{\mu}+r \end{smallmatrix}\right]_\gamma \leftarrow \mathsf{ek} \cdot [\mathsf{c}]_\gamma$; **return** $[\boldsymbol{M}\boldsymbol{\mu}]_\gamma$.

Fig. 3. The $[\cdot]_\gamma$-FSE commitment scheme FSE_γ for linear maps in $\mathbb{G}_\gamma$.

3.2 QA-NIZK

A QA-NIZK argument system [40] Π has public parameters lp (a language parameter, like a commitment key) and crs (a language-dependent common reference string). Π proves membership in the language $\mathcal{L}_{\mathsf{lp}}$ defined by a relation $\mathcal{R}_{\mathsf{lp}} = \{(\mathbb{x}, \mathbb{w})\}$. Both are determined by $\mathsf{lp} \leftarrow\!\$\ \mathcal{D}_{\mathrm{par}}$ (sampled by PPT K_{lp}), where $\mathcal{D}_{\mathrm{par}}$ is a public distribution. $\mathcal{D}_{\mathrm{par}}$ is *witness-sampleable* [40] if there exists a PPT algorithm K_{lt} that samples $(\mathsf{lp}, \mathsf{lt})$ such that lp is distributed according to $\mathcal{D}_{\mathrm{par}}$, and $\mathsf{lp} \in^? \mathrm{range}(\mathcal{D}_{\mathrm{par}})$ can be efficiently verified given lt.

A *QA-NIZK* for $\mathcal{R}_{\mathrm{par}}$ is a tuple of PPT algorithms $\Pi = (\mathsf{Pgen}, \mathsf{K}_{\mathsf{lp}}, \mathsf{K}_{\mathsf{crs}}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$. In the case of witness-sampleable languages, K_{lp} is replaced by K_{lt}. Pgen is the parameter generation algorithm, K_{lp} is the language parameter generation algorithm, K_{lt} is the corresponding generation algorithm in the witness-sampleable case that creates lp and lt, $\mathsf{K}_{\mathsf{crs}}$ is the CRS generation algorithm, P is the prover, V is the verifier, and Sim is the simulator. We assume that lp contains p. Sim is a single algorithm that works for each relation in $\mathcal{R}_{\mathrm{par}} := \{\mathcal{R}_{\mathsf{lp}}\}_{\mathsf{lp}\in\mathrm{range}(\mathcal{D}_{\mathrm{par}})}$.

Π can satisfy the following security notions.

Perfect Completeness: for all λ and PPT $\mathcal{A}$,

$$\Pr\left[\begin{array}{l|l} \mathsf{V}(\mathsf{lp},\mathsf{crs},\mathbb{x},\pi) = 0 \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); \\ (\mathbb{x},\mathbb{w}) \in \mathcal{R}_{\mathsf{lp}} & (\mathsf{crs},\mathsf{td}) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}); (\mathbb{x},\mathbb{w}) \leftarrow \mathcal{A}(\mathsf{lp},\mathsf{crs}); \\ & \pi \leftarrow \mathsf{P}(\mathsf{lp},\mathsf{crs},\mathbb{x},\mathbb{w}) \end{array}\right] = 0 \ .$$

Computational Quasi-Adaptive Strong Soundness: defined only if lp is witness-sampleable. For any PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{strsound}}_{\mathsf{Pgen},\mathcal{D}_{\mathrm{par}},\mathrm{BLS},\mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|l} \mathsf{V}(\mathsf{lp},\mathsf{crs},\mathbb{x},\pi) = 1 \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); (\mathsf{lp},\mathsf{lt}) \leftarrow \mathsf{K}_{\mathsf{lt}}(\mathsf{p}); \\ (\neg\exists \mathbb{w})(\mathbb{x},\mathbb{w}) \in \mathcal{R}_{\mathsf{lp}} & (\mathsf{crs},\mathsf{td}) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}); (\mathbb{x},\pi) \leftarrow \mathcal{A}(\mathsf{lp},\mathsf{lt},\mathsf{crs}) \end{array}\right] \approx_\lambda 0 \ .$$

Perfect Zero Knowledge: for all unbounded $\mathcal{A}$, $|\varepsilon_1^{zk} - \varepsilon_2^{zk}| = 0$, where $\varepsilon_\beta^{zk} :=$

$$\Pr\left[\, \mathcal{A}^{\mathcal{O}_\beta(\cdot,\cdot)}(\mathsf{lp},\mathsf{crs}) = 1 \,\middle|\, \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); (\mathsf{crs},\mathsf{td}) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}) \,\right] \ .$$

Here, $\mathcal{A}$ is given an oracle access to $\mathcal{O}_\beta(\cdot,\cdot)$, where $\mathcal{O}_0(\mathbb{x},\mathbb{w})$ returns 0 (reject) if $(\mathbb{x},\mathbb{w}) \notin \mathcal{R}_{\mathsf{lp}}$, and otherwise it returns $\mathsf{P}(\mathsf{lp},\mathsf{crs},\mathbb{x},\mathbb{w})$. Similarly, $\mathcal{O}_1(\mathbb{x},\mathbb{w})$ returns 0 if $(\mathbb{x},\mathbb{w}) \notin \mathcal{R}_{\mathsf{lp}}$, and otherwise it returns $\mathsf{Sim}(\mathsf{lp},\mathsf{crs},\mathsf{td},\mathbb{x})$.

C&P QA-SNARGs. A QA-NIZK is *succinct* (succinct non-interactive argument, QA-SNARG) if the argument length is sublinear in $\mathsf{poly}(\lambda) \cdot (|\mathbb{x}| + |\mathbb{w}|)$. It is commit-and-prove (C&P) if ck is a commitment key and the statement contains an extractable commitment (to a witness) under this commitment key.

Gentry-Wichs Impossibility Result. Gentry and Wichs [26] proved that if an NP language $\mathcal{L}$ has a sub-exponentially (resp., exponentially) hard subset-membership problem and Π is a complete SNARG in the CRS model with $|\pi| \leq \mathsf{poly}(\lambda) \cdot (|\mathbb{x}| + |\mathbb{w}|)^{o(1)}$ (resp., $|\pi| \leq \mathsf{poly}(\lambda) \cdot (|\mathbb{x}| + |\mathbb{w}|)^c + o(|\mathbb{x}| + |\mathbb{w}|)$ for some constant $c < 1$) for $\mathcal{L}$, then there is a black-box reduction from the adaptive soundness of Π to a falsifiable assumption X only when X is false. [12] clarifies why linear subspace QA-SNARGs do not contradict Gentry-Wichs. In Appendix A.1, we explain how this relates to the current work.

Bilateral Subspace QA-SNARG. Denote $[\boldsymbol{M}]_* := ([\boldsymbol{M}_1]_1, [\boldsymbol{M}_2]_2)$. A bilateral subspace argument system, with $\mathsf{ck} = [\boldsymbol{M}]_* \in \mathbb{G}_1^{n_1 \times m} \times \mathbb{G}_2^{n_2 \times m}$, allows to prove that $([\mathbf{c}_1]_1, [\mathbf{c}_2]_2) \in \mathcal{L}_{\mathsf{ck}}$, where

$$\mathcal{L}_{\mathsf{ck}} := \left\{ ([\mathbf{c}_1]_1, [\mathbf{c}_2]_2) \in \mathbb{G}_1^{n_1} \times \mathbb{G}_2^{n_2} : (\exists \mathbb{w} \in \mathbb{F}^m) \left(\begin{smallmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{smallmatrix}\right) = \left(\begin{smallmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \end{smallmatrix}\right) \mathbb{w} \right\} ,$$

that is, $\left(\begin{smallmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{smallmatrix}\right) \in \mathrm{colspace}\left(\begin{smallmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \end{smallmatrix}\right)$. Note that it does not have the C&P property, unless $[\boldsymbol{M}]_*$ is a commitment key.

For the sake of completeness, in Fig. 8 (see Appendix B.1), we depict the González-Hevia-Ràfols bilateral subspace QA-SNARG argument system BLS for $\mathcal{L}_{\mathsf{ck}}$. Lipmaa and Pavlyk [52] generalized a theorem by González and Ràfols [27] to any $n_\gamma \times m$ matrices $\boldsymbol{M}_\gamma$ (even if $m > n_\gamma$), given that $\mathrm{rank}\left(\begin{smallmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \end{smallmatrix}\right) < n_1 + n_2$. This generalization is important for us since in Punic (see Eq. (4)), $m > n_1, n_2$.

Fact 2 ([27,52]) *Fix λ, n_1, n_2, m. Let $\kappa = 2$. Let $\mathcal{D}_{\mathrm{par}}$ be a matrix distribution on $[\boldsymbol{M}]_* \in \mathbb{G}_1^{n_1 \times m} \times \mathbb{G}_2^{n_2 \times m}$, such that $\mathrm{rank}\left(\begin{smallmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \end{smallmatrix}\right) < n_1 + n_2$. Then (1)* BLS *is perfectly complete and perfectly zero-knowledge. (2) Assume $\mathcal{D}_{\mathrm{par}}$ is witness-sampleable and $\mathcal{D}_\kappa$ is robust. If $\mathcal{D}_\kappa$-SKerMDH holds relative to Pgen then* BLS *is computationally quasi-adaptively strongly sound.*

We need $\kappa = 2$ since SKerMDH does not hold for $\kappa = 1$ [27]. The prover's work is dominated by $2m\kappa$ scalar multiplications, the verifier's work is dominated by $(n_1 + n_2 + 2\kappa)\kappa$ pairings, and π consists of 2κ group elements.

4 New Vector Commitment Scheme

We need a pairing-based vector commitment scheme VCF that is fully-algebraic and F-position-binding. Since we use the [19]'s FSE to black-box extract VCF's local openings and local proofs, both novel requirements are needed. W.l.o.g., we consider commitment schemes with an output from $\mathbb{G}_1$.

4.1 Definitions

Fully-Algebraic. Recall that a commitment scheme is *algebraic* if $\mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r) = [\boldsymbol{M}^*]_1(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$ for a matrix $[\boldsymbol{M}^*]_1$ efficiently computable from ck.

Definition 2. *A vector commitment scheme is* fully-algebraic, *if* $C := \mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r) = [\boldsymbol{M}^*]_1(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$, $[\eta]_1 = [\boldsymbol{M}_j^\eta]_1(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$, *and* $[\varphi]_1 = [\boldsymbol{M}_j^\varphi]_1(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$, *where* $[\eta, \varphi]_1 = \mathsf{LOpen}(\mathsf{ck}, C, j, (\boldsymbol{\mu}, r))$, *for some public matrices* $[\boldsymbol{M}^*]_1$, $[\boldsymbol{M}_j^\eta]_1$, *and* $[\boldsymbol{M}_j^\varphi]_1$ *that can be efficiently computed from* ck *and (in the last two cases)* j.

Let $\boldsymbol{e}_j$ be the jth unit vector,. Clearly, $[\eta]_1 = [\mu_j]_1 = [\boldsymbol{e}_j^\intercal \| 0]_1 \cdot (\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$ holds for any vector commitment scheme. Thus, the existence of $\boldsymbol{M}_j^\eta$ is trivial and one needs to only show $\mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r) = [\boldsymbol{M}^*]_1(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$ and $[\varphi]_1 = [\boldsymbol{M}_j^\varphi]_1(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix})$.

The vector commitment scheme of Catalano and Fiore [13] is fully algebraic, but it has a commitment key of $\Theta(n^2)$ group elements and is thus inefficient. The more efficient vector commitment scheme of Libert et al. [47,36] is not fully algebraic. The CDHK [10] vector commitment scheme is efficient and algebraic but not known to be fully algebraic. In Section 4.2, we show that CDHK is fully algebraic. However, it does not satisfy the following requirement.

***F*-Position-Binding.** In Punic, we use FSE to black-box extract $F(\eta) = F(\mu_j)$ for a one-way permutation F. Thus, we need the vector commitment scheme to be position-binding even if the position-binding adversary outputs $F(\eta)$ instead of η. This is similar to how F-unforgeable signature schemes [5] is defined when the adversary outputs $F(\boldsymbol{\mu})$ instead of the message $\boldsymbol{\mu}$. F-position-binding suffices in our case since in the soundness proof of Punic, we are not interested in the value of η but only in testing whether two local openings are equal. Since F is a permutation, such testing can be performed on $F(\eta)$ and $F(\eta')$.

Definition 3. *An* F-position-binding vector commitment scheme *is a commitment scheme that has the following additional algorithms:*

Local F-opening: *for* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathsf{ck} \in \mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n)$, *a commitment* $C \in \mathcal{C}$, *a coordinate* $j \in [1, n]$, *and a decommitment information* D, $\mathsf{LOpen}_F(\mathsf{ck}, C, j, D)$ *returns* $(F(\eta), \varphi)$, *where* η *is a local opening (a candidate value of* μ_j*) and* φ *is a local proof.*

Local F-verification: *for* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathsf{ck} \in \mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n)$, *a commitment* $C \in \mathcal{C}$, *a coordinate* $j \in [1, n]$, *a local opening* $F(\eta)$, *and a local proof* φ, $\mathsf{LVer}_F(\mathsf{ck}, C, j, F(\eta), \varphi)$ *returns either* 0 *or* 1.

It must be complete and satisfy the following security notion:

F-position-binding: *for all* λ, *PPT* $\mathcal{A}$, *and* $n \in \mathsf{poly}(\lambda)$, $\mathsf{Adv}^{\mathrm{fposb}}_{\mathsf{Pgen},F,n,\Gamma,\mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|l} \eta_0 \neq \eta_1 \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); (\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n); \\ \mathsf{LVer}_F(\mathsf{ck}, C, j, F(\eta_0), \varphi_0) = 1 \wedge & (C, j, F(\eta_0), F(\eta_1), \varphi_0, \varphi_1) \leftarrow \mathcal{A}(\mathsf{ck}) \\ \mathsf{LVer}_F(\mathsf{ck}, C, j, F(\eta_1), \varphi_1) = 1 & \end{array}\right]$$

is negligible.

We will omit the subscript F when it is clear from the context. In Punic, F is such that the FSE commitment scheme is somewhere F-extractable. In the case of the FSE commitment scheme of [19], $F = [\cdot]_1$ or $F = [\cdot]_2$.

$\mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n)$: $x \leftarrow\!\!\$\ \mathbb{F} \setminus \mathbb{H}$; $y \leftarrow\!\!\$\ \mathbb{F}^*$; $\mathsf{td} \leftarrow (x, y)$;
$\mathsf{ck} \leftarrow ([(x^i)_{i=0}^n, y]_\gamma, [(x^i)_{i=0}^n, y, xy]_{3-\gamma})$; // Private verif.: $[xy]_\gamma \notin \mathsf{ck}$
compute and store $[\mathsf{ck}_\ell]_\gamma \leftarrow [\ell_1(x), \ldots, \ell_n(x), Z_{\mathbb{H}}(x)]_\gamma$;
return $(\mathsf{ck}, \mathsf{td})$;
$\mathsf{Com}(\mathsf{ck}, \boldsymbol{\mu}; r)$: $r \leftarrow\!\!\$\ \mathbb{F}$; $[C(x)]_\gamma \leftarrow [\mathsf{ck}_\ell]_\gamma \cdot \binom{\boldsymbol{\mu}}{r} = \sum_{i=1}^n \mu_i [\ell_i(x)]_\gamma + r[Z_{\mathbb{H}}(x)]_\gamma$;
return $([C(x)]_\gamma, (\boldsymbol{\mu}, r))$; // (C, D)
$\mathsf{LOpen}(\mathsf{ck}, [C(x)]_\gamma, j, (\boldsymbol{\mu}, r))$: $[\hat{\eta}]_\gamma \leftarrow \mu_j [y]_\gamma$; // $\mu_j = C(\omega^{j-1})$
$[\mathsf{ck}_{\ell,j}]_\gamma \leftarrow [Q_{\ell_1,j}(x), \ldots, Q_{\ell_n,j}(x), Q_{Z_{\mathbb{H}},j}(x)]_\gamma$;
$[\varphi(x)]_\gamma \leftarrow [\mathsf{ck}_{\ell,j}]_\gamma \cdot \binom{\boldsymbol{\mu}}{r} = \sum_{i=1}^n \mu_i [Q_{\ell_i,j}(x)]_\gamma + r[Q_{Z_{\mathbb{H}},j}(x)]_\gamma$;
return $[\hat{\eta}, \varphi(x)]_\gamma$.
$\mathsf{LVer}(\mathsf{ck}, [C(x)]_\gamma, j, [\hat{\eta}, \varphi(x)]_\gamma)$: check $[C(x)]_\gamma \bullet [y]_{3-\gamma} - [\hat{\eta}]_\gamma \bullet [1]_{3-\gamma} = [\varphi(x)]_\gamma \bullet ([xy]_{3-\gamma} - \omega^{j-1}[y]_{3-\gamma})$; // Public verification only
$\mathsf{Sim}(\mathsf{ck}, \mathsf{td} = x, \{j_i\}_{i\in I}, \{\mu_{j_i}\}_{i\in I})$: $r \leftarrow\!\!\$\ \mathbb{F}$; $r' \leftarrow (\sum_{i\in I} \mu_{j_i} \ell_{j_i}(x))/Z_{\mathbb{H}}(x) + r$; return $[C(x)]_\gamma \leftarrow \mathsf{Com}(\mathsf{ck}, \mathbf{0}; r') = r'[Z_{\mathbb{H}}(x)]_\gamma$;

Fig. 4. The new $[\cdot]_\gamma$-position-binding vector commitment scheme VCF_γ.

4.2 Construction

CDHK is clearly algebraic. We will show that it is fully algebraic by showing that $[\varphi(x)]_1$ can be computed by using a linear map.

For a polynomial $f(X) \in \mathbb{F}[X]$ and an integer $j \in [1, n]$, let $Q_{f,j}(X)$ be the quotient of $(f(X) - f(\omega^{j-1}))/(X - \omega^{j-1})$. Clearly, $\deg Q_{f,j} = \deg f - 1$.

Lemma 1. *Fix $j \in [1, n]$. For $C(X) = \sum_{i=1}^n \mu_i \ell_i(X) + rZ_{\mathbb{H}}(X) \in \mathbb{F}[X]$, $Q_{C,j}(X) = \frac{C(X) - C(\omega^{j-1})}{X - \omega^{j-1}}$. Then, $[Q_{C,j}(x)]_1 = [\mathsf{ck}_{\ell,j}(x)]_1 \cdot \binom{\boldsymbol{\mu}}{r}$, where $\mathsf{ck}_{\ell,j}(X) := (Q_{\ell_1,j}(X), \ldots, Q_{\ell_n,j}(X), Q_{Z_{\mathbb{H}},j}(X))$. Thus,* CDHK *is fully-algebraic.*

Proof. Clearly, $Q_{C,j}(X) = (C(X) - C(\omega^{j-1}))/(X - \omega^{j-1})$ is equal to $(\sum_{i=1}^n \mu_i \ell_i(X) + rZ_{\mathbb{H}}(X) - \mu_j)/(X - \omega^{j-1})$. Since $\sum_{i=1}^n \mu_i \ell_i(\omega^{j-1}) = \mu_j$ and $Z_{\mathbb{H}}(\omega^{j-1}) = 0$, $Q_{C,j}(X) = \sum_{i=1}^n \mu_i Q_{\ell_i,j}(X) + rQ_{Z_{\mathbb{H}},j}(X) = \mathsf{ck}_{\ell,j}(X) \cdot \binom{\boldsymbol{\mu}}{r}$. □

Making CDHK $[\cdot]_\gamma$-Position-Binding. One can easily break $[\cdot]_\gamma$-position-binding of CDHK (see Fig. 2) by outputting $([C]_\gamma, j, [\eta, \eta']_\gamma, [\varphi, \varphi']_\gamma)$, where $[C]_\gamma = [x - \omega^{j-1}]_\gamma$, $[\eta]_\gamma = [0]_\gamma$, $[\eta']_\gamma = [x - \omega^{j-1}]_\gamma$, $[\varphi]_\gamma = [1]_\gamma$, and $[\varphi']_\gamma = [0]_\gamma$. Clearly, $C - \eta = \varphi(x - \omega^{j-1})$ and $C - \eta' = \varphi'(x - \omega^{j-1})$.

We avoid such attacks by guaranteeing that $[\eta, \eta']_\gamma$ do not depend on x. We achieve this by making the local opening depend on a new trapdoor y and not adding $[x^i y]_\gamma$ to ck for $i > 0$. (However, $[y]_1, [y]_2, [y, xy]_{3-\gamma}$ must be in ck for VCF to be publicly verifiable.) Importantly, the communication does not increase. In Fig. 4, we depict the new vector commitment scheme VCF_γ. Clearly, $C(x) = (x - \omega^{j-1})\varphi(x) + \hat{\eta}/y$ since the remainder of $\ell_i(X)/(X - \omega^{j-1})$ is 1 if $i = j$ and 0, otherwise. The local opening is $G_\gamma(\mu_j) = G_\gamma(\eta)$ for $G_\gamma(s) := [sy]_\gamma$.

The soundness proofs (but not the constructions) of QA-SNARGs of [18,19] use *implicitly* a version of VCF but without defining the used primitive as a

vector commitment scheme or writing down the needed security properties. Their implicit vector commitment scheme is less efficient, requiring the local opening to output both $\mu_j[1]_\gamma$ and $\mu_j[y]_\gamma$. Their constructions also use a perfectly-hiding commitment scheme, while we use only VCF.

Private-Verifiability. Punic uses both VCF_1 and VCF_2. We need to use the same trapdoor in both cases, and thus want to have the same ck when defining VCF_γ. Thus, although this is not necessary for VCF_γ itself, we add $[(x^i)_{i=0}^n]_{3-\gamma}$ to the commitment key. However, we cannot add $[xy]_\gamma$ to ck since that would break VCF's security. To overcome this, one possibility is to reuse the trapdoor x but have separate trapdoors y_1 and y_2 in VCF_1 and VCF_2. We opted for a simpler possibility: since in Punic, VCF_γ does not have to be publicly verifiable, one can omit $[xy]_\gamma$ (only used in verification) from ck. This allows us to reuse the same trapdoor y in both VCF_1 and VCF_2. From now on, we will always use the privately verifiable version of VCF_γ with $\mathsf{ck} \leftarrow ([(x^i)_{i=0}^n, y]_\gamma, [(x^i)_{i=0}^n, y]_{3-\gamma})$.

4.3 Security Analysis

VCF_γ is clearly perfectly zero-knowledge. From a position-binding collision $([C(x)]_\gamma, j, [\hat{\eta}, \hat{\eta}']_\gamma, [\varphi, \varphi']_\gamma)$ with $\hat{\eta} \neq \hat{\eta}'$, we get $[\hat{\eta}' - \hat{\eta}]_\gamma \bullet [1]_{3-\gamma} = [\varphi - \varphi']_\gamma \bullet [(x - \omega^{j-1})y]_{3-\gamma}$ and thus $[\varphi - \varphi']_\gamma = \frac{1}{(x-\omega^{j-1})y}[\hat{\eta}' - \hat{\eta}]_\gamma$. We define a new assumption n-VCSDH that states that it is difficult to output $[\varphi - \varphi']_\gamma$ and $[\hat{\eta}' - \hat{\eta}]_\gamma \neq [0]_\gamma$ that satisfy the above equation.

Definition 4. n-Vector-Commitment Strong Diffie-Hellman (n-VCSDH) *holds relative to* Pgen *in* $\mathbb{G}_\gamma$*, if for all PPT* $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{vcsdh}}_{\mathsf{Pgen},\gamma,n,\mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|l} \hat{\eta} \neq 0 \wedge & \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); x \leftarrow\!\$\, \mathbb{F} \setminus \mathbb{H}; y \leftarrow\!\$\, \mathbb{F}^*; \\ [\varphi]_\gamma = \frac{1}{(x-\omega^{j-1})y}[\hat{\eta}]_\gamma & \mathsf{ck} \leftarrow ([(x^i)_{i=0}^n, y]_1, [(x^i)_{i=0}^n, y]_2); \\ & (j, [\hat{\eta}, \varphi]_\gamma) \leftarrow \mathcal{A}(\mathsf{ck}) \end{array}\right] \approx_\lambda 0 \ .$$

The following lemma is straightforward.

Lemma 2. *Privately-verifiable* VCF_1 *is* $[\cdot]_\gamma$*-position-binding iff* n-VCSDH *holds relative to* Pgen.

VCSDH is similar to known SDH-like [9] assumptions like RSDH [28]. VCSDH is privately-verifiable but clearly falsifiable. It is intuitively secure since $[\hat{\eta}]_\gamma$ cannot depend on xy, and thus $\varphi(x, y)$ is not a polynomial. Next, prove that VCSDH follows from QALINRES, which was proven in [52] to be secure in the AGM, [22]. Thus, VCSDH is secure in the AGM and falsifiable. Punic relies on QALINRES and not on VCSDH directly.

Lemma 3. *Fix* $n = \mathsf{poly}(\lambda)$*. If* n-QALINRES *holds, then* n-VCSDH *holds.*

See Appendix D.1 for the proof.

QALINRES can restated as an algebraic security property of privately-verifiable VCF_γ, observing that say $\mathsf{a} = \varphi_\mathsf{a}(x - \omega^{j-1}) + \hat{\eta}_\mathsf{a}/y$ iff $\mathrm{VCF}_1.\mathsf{LVer}(\mathsf{ck}, [\mathsf{a}]_1, j, [\hat{\eta}_\mathsf{a}]_1, [\varphi_\mathsf{a}]_1)$. Privately-verifiable VCF_1 and VCF_2 share the commitment key; this is possible since we do not require QALINRES to be publicly-verifiable.

4.4 Committing to Linear Maps

We need the following result. See Appendix D.2 for the proof.

Lemma 4. *Let* VCF_1 *be as in Fig. 4 Let* $\boldsymbol{\mu} \in \mathbb{F}^m$ *and* $\boldsymbol{U} \in \mathbb{F}^{n\times m}$. *Let* $u_j(X) := \sum_{i=1}^{n} U_{ij}\ell_i(X)$ *be the interpolating vector of* $\boldsymbol{U}^{(j)}$, $\mathsf{ck}_u := (u_1(x)\|\ldots\|u_m(x)\|Z_{\mathbb{H}}(x)) = \mathsf{ck}_\ell \cdot \left(\begin{smallmatrix}\boldsymbol{U} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{smallmatrix}\right)$, $[\mathsf{ck}_{\boldsymbol{e},j}]_1 := G(\boldsymbol{e}_j^\intercal\|0)$, *and* $\mathsf{ck}_{u,j} := (Q_{u_1,j}(x)\|\ldots\|Q_{u_m,j}(x)\|Q_{Z_{\mathbb{H}},j}(x))$. *Then,* $[C(x)]_1 \leftarrow \mathsf{Com}(\mathsf{ck}, \boldsymbol{U}\boldsymbol{\mu}; r)$ *and* $(G(\eta), [\varphi]_1) \leftarrow \mathsf{LOpen}(\mathsf{ck}, [C(x)]_1, j, D = (\boldsymbol{U}\boldsymbol{\mu}, r))$ *are linear maps of* $\left(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix}\right)$*:*

$$[C(x)]_1 = [\mathsf{ck}_u]_1 \cdot \left(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix}\right) \ , \quad G(\eta) = G(\mathsf{ck}_{\boldsymbol{e},j}) \cdot \left(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix}\right) \ , \quad [\varphi]_1 = [\mathsf{ck}_{u,j}]_1 \cdot \left(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix}\right) \ .$$

Thus, one can compute the commitment to $\boldsymbol{U}\boldsymbol{\mu}$ and its local proof as $[\mathsf{ck}_u]_1\left(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix}\right)$ and $[\mathsf{ck}_{u,j}]_1\left(\begin{smallmatrix}\boldsymbol{\mu}\\ r\end{smallmatrix}\right)$ given public matrices that depend on x, $\boldsymbol{U}$, and j.

5 New C&P Zk-SNARK Security Notions

The new C&P zk-SNARK satisfies a novel soundness notion, semi-adaptive black-box G-knowledge-soundness. As motivated in Section 2.3, semi-adaptivity is a natural version of non-adaptivity for C&P SNARKs. Black-box G-knowledge-soundness is stronger than local consistency (Kalai et al., [41]). Semi-adaptive black-box G-knowledge-soundness is a semi-adaptive variant of the non-adaptive black-box knowledge-soundness of [12]. Moreover, we need Punic to be CRS-indistinguishable. Next, we define the new security notions.

5.1 R1CS And $\mathrm{R1CS}_{\mathfrak{f}}$

Let n be the number of constraints, m be the number of variables, and $m_{\mathbb{x}} < m$ be the number of public inputs and outputs. Let $\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W} \in \mathbb{F}^{n\times m}$ be instance-dependent matrices and let $\left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right) \in \mathbb{F}^m$. An R1CS [25] instance $\mathfrak{J} = (\mathbb{F}, m_{\mathbb{x}}, \boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W})$ defines the following relation[5]:

$$\mathcal{R}_{\mathfrak{J}} = \left\{(\mathbb{x}, \mathbb{w}) : \mathbb{x} \in \mathbb{F}^{m_{\mathbb{x}}} \ \wedge \ \mathbb{w} \in \mathbb{F}^{m-m_{\mathbb{x}}} \ \wedge \ \boldsymbol{U}\left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right) \circ \boldsymbol{V}\left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right) = \boldsymbol{W}\left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right)\right\} \ . \quad (1)$$

We say $(\mathbb{x}, \mathbb{w})$ *satisfies* $\mathfrak{J}$ if $(\mathbb{x}, \mathbb{w}) \in \mathcal{R}_{\mathfrak{J}}$. Crucially, one can check that $(\mathbb{x}, \mathbb{w}) \in \mathcal{R}_{\mathfrak{J}}$ by checking a conjugation of local constraints. For a constraint $\varrho \in [1, n]$,

$$N_{\mathfrak{J}}(\varrho) := \{j : U_{\varrho j} \neq 0 \vee V_{\varrho j} \neq 0 \vee W_{\varrho j} \neq 0\}$$

(the *ϱth neighborhood*) is the set of variables in the neighborhood of the constraint ϱ. We usually omit the subscript $\mathfrak{J}$. W.l.o.g., assume that the set of neighborhoods covers the whole range $[1, m]$. Otherwise, some variables are not used in the instance and can thus be omitted. For $\mathfrak{f} \geq 1$, let $\mathrm{R1CS}_{\mathfrak{f}}$ be the language of instances $\mathfrak{J}$, such that $|N(\varrho)| \leq \mathfrak{f}$ for all ϱ.

Fix $\varrho \in [1, n]$. Let $\mathbb{p}^\varrho : N(\varrho) \to \mathbb{F}$ be an assigment of variables from $N(\varrho)$. We say that $(\mathbb{x}, \mathbb{p}^\varrho)$ *locally satisfies* the instance $\mathfrak{J}$ iff

[5] $(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W})$ is a part of the instance and thus our SNARKs are non-universal. The most efficient known universal SNARKs [32] in the standard model (without random oracles) have quadratic size CRS and are thus too inefficient for practice.

(1) $\mathbb{p}^{\varrho}$ agrees with the statement $\mathbb{x}$: $(\forall j \in ([1, m_{\mathbb{x}}] \cap N(\varrho)))\, \mathbb{p}^{\varrho}(j) = \mathbb{x}_j$, and
(2) $\mathbb{p}^{\varrho}$ satisfies the ϱth constraint:

$$(\textstyle\sum_{j \in N(\varrho)} U_{\varrho j} \mathbb{p}^{\varrho}(j)) \cdot (\sum_{j \in N(\varrho)} V_{\varrho j} \mathbb{p}^{\varrho}(j)) = \sum_{j \in N(\varrho)} W_{\varrho j} \mathbb{p}^{\varrho}(j) \ .$$

If only 1 holds, we say that $\mathbb{p}^{\varrho}$ satisfies the ϱth constraint. If both 1 and 2 hold, we write $(\mathbb{x}, \mathbb{p}^{\varrho}) \in \mathcal{R}^{\varrho}_{\mathsf{loc},\mathfrak{J}}$, where $\mathcal{R}^{\varrho}_{\mathsf{loc},\mathfrak{J}} :=$

$$\left\{ (\mathbb{x}, \mathbb{p}^{\varrho}) \middle| \begin{array}{l} ((\forall j \in ([1, m_{\mathbb{x}}] \cap N(\varrho)))\, \mathbb{p}^{\varrho}(j) = \mathbb{x}_j) \wedge \\ ((\sum_{j \in N(\varrho)} U_{\varrho j} \mathbb{p}^{\varrho}(j)) \cdot (\sum_{j \in N(\varrho)} V_{\varrho j} \mathbb{p}^{\varrho}(j)) = (\sum_{j \in N(\varrho)} W_{\varrho j} \mathbb{p}^{\varrho}(j))) \end{array} \right\} . \quad (2)$$

Note that the second element of $(\mathbb{x}, \mathbb{w}) \in \mathcal{R}_{\mathfrak{J}}$ is a full witness while the second element of $(\mathbb{x}, \mathbb{p}^{\varrho}) \in \mathcal{R}^{\varrho}_{\mathsf{loc},\mathfrak{J}}$ is a partial witness. Moreover, one can use pairings to check $(\mathbb{x}, \mathbb{p}^{\varrho}) \in^{?} \mathcal{R}^{\varrho}_{\mathsf{loc},\mathfrak{J}}$ even if only given $(\mathbb{x}, [\mathbb{p}^{\varrho}]_1)$.

For $i, j \in [1, n]$, we define the *consistency predicate*

$$\mathsf{Cons}(\mathbb{p}^i, \mathbb{p}^j) := (\forall k \in (N(i) \cap N(j)))\, \mathbb{p}^i(k) = \mathbb{p}^j(k) \ ,$$

Remark 1. Fix $\mathbb{x}$. Clearly, there exists a full witness $\mathbb{w} \in \mathcal{R}_{\mathfrak{J}}$ that satisfies all constraints and agrees with all partial assignments $\mathbb{p}^{\varrho}$ if
(1) for each constraint ϱ, $(\mathbb{x}, \mathbb{p}^{\varrho})$ is locally satisfied,
(2) for all constraints i, j, $\mathsf{Cons}(\mathbb{p}^i, \mathbb{p}^j) = \mathsf{true}$.

Fix a commitment scheme and instance $\mathfrak{J}$. We assume the statement is $\mathbb{x}^{\dagger} := (C, \mathbb{x})$ and the witness is $\mathbb{w}^{\dagger} := (r_C, \mathbb{w})$, where C is a commitment and r_C is a commitment randomness. For a fixed $\mathsf{lp} = \mathsf{ck}$, we define

$$\mathcal{R}_{\mathsf{lp}} := \{((C, \mathbb{x}), (r_C, \mathbb{w})) : C = \mathsf{Com}\,((\tfrac{\mathbb{x}}{\mathbb{w}}); r_C) \wedge (\mathbb{x}, \mathbb{w}) \in \mathcal{R}_{\mathfrak{J}}\}$$

to be the relation $\mathcal{R}_{\mathsf{lp}}$ from Section 3.2.

5.2 Security Definitions

We redefine C&P zk-SNARKs for $\mathrm{R1CS}_{\mathfrak{f}}$ allowing $\mathsf{K}_{\mathsf{crs}}$ to depend on a constraint number ϱ, where an honest execution sets $\varrho \leftarrow 0$ while the reductions use non-zero ϱ's. (An alternative approach is to define two different $\mathsf{K}_{\mathsf{crs}}$'s.) Fix a (vector) commitment scheme Γ. Then, $\mathsf{lp} = \mathsf{ck}$ is a commitment key. We also assume that there exists a black-box somewhere-extractor $\mathsf{Ext}_{\mathsf{ks}}$.

The modified (QA-)SNARK security definitions follow. We highlight the changes to the definition in Section 3.2. We require that completeness holds for all choices of ϱ while zero-knowledge holds for the value of ϱ, $\varrho = 0$, used in the honest case. Computational zero-knowledge for *any* ϱ follows from this and the CRS-indistinguishability. The latter (see Definition 5) guarantees that the CRSs corresponding to different constraints are computationally indistinguishable.

Perfect Completeness: for all λ, PPT $\mathcal{A}$, and $\varrho \in [1, n]$,

$$\Pr\left[\begin{array}{l} \mathsf{V}(\mathsf{lp}, \mathsf{crs}, \mathbb{x}^{\dagger}, \pi) = 0 \\ \wedge\, (\mathbb{x}, \mathbb{w}) \in \mathcal{R}_{\mathfrak{J}} \end{array} \middle| \begin{array}{l} \mathsf{p} \leftarrow \mathsf{Pgen}(1^{\lambda}); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); \\ (\mathsf{crs}, \mathsf{td}) \leftarrow\!\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho); \\ (\mathbb{x}, \mathbb{w}, r_C) \leftarrow \mathcal{A}(\mathsf{lp}, \mathsf{crs}); C \leftarrow \mathsf{Com}((\tfrac{\mathbb{x}}{\mathbb{w}}); r_C); \\ \mathbb{x}^{\dagger} \leftarrow (C, \mathbb{x}); \mathbb{w}^{\dagger} \leftarrow (r_C, \mathbb{w}); \pi \leftarrow \mathsf{P}(\mathsf{lp}, \mathsf{crs}, \mathbb{x}^{\dagger}, \mathbb{w}^{\dagger}) \end{array} \right] = 0 \ .$$

Perfect Zero Knowledge: for all unbounded $\mathcal{A}$, $|\varepsilon_1^{zk} - \varepsilon_2^{zk}| = 0$, where $\varepsilon_\beta^{zk} :=$

$$\Pr\left[\mathcal{A}^{\mathcal{O}_\beta(\cdot,\cdot)}(\mathsf{lp},\mathsf{crs}) = 1 \middle| \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); (\mathsf{crs},\mathsf{td}) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},0)\right] .$$

$\mathcal{A}$ is given an oracle access to $\mathcal{O}_\beta(\cdot,\cdot)$, where $\mathcal{O}_0(\mathbb{x}^\dagger,\mathbb{w}^\dagger)$ returns 0 if $(\mathbb{x}^\dagger,\mathbb{w}^\dagger) \notin \mathcal{R}_{\mathsf{lp}}$; otherwise, it returns $\mathsf{P}(\mathsf{lp},\mathsf{crs},\mathbb{x}^\dagger,\mathbb{w}^\dagger)$. Similarly, $\mathcal{O}_1(\mathbb{x}^\dagger,\mathbb{w}^\dagger)$ returns 0 if $(\mathbb{x}^\dagger,\mathbb{w}^\dagger) \notin \mathcal{R}_{\mathsf{lp}}$; otherwise, it returns $\mathsf{Sim}(\mathsf{lp},\mathsf{crs},\mathsf{td},\mathbb{x}^\dagger)$.

We define a new knowledge soundness notion that has two aspects. First, *semi-adaptivity*. In the quasi-adaptive case, the statement can depend on lp and crs, while in the semi-adaptive case, it can only depend on lp. Second, in local consistency [55,41,29] it is required that, given $\mathsf{crs}^\varrho \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},\varrho)$, one can black-box somewhere-extract a partial witness that satisfies the ϱth constraint. We strengthen this by requiring one to black-box extract a full witness that satisfies all constraints.

Definition 5 is inspired by non-adaptive black-box knowledge-soundness in [12] and witness-extended emulation (WEE, [48]). Let G be a permutation. Definition 5 formalizes our expected ability to black-box extract $G(\mathbb{w})$, where $\mathbb{w}$ satisfies all constraints, by running the adversary with many different CRSs crs^ϱ, where crs^ϱ is output by $\mathsf{K}_{\mathsf{crs}}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},\varrho)$, and then gluing the adversary's outputs to $G(\mathbb{w})$. We relate the probability that an adversary outputs an accepting transcript to the probability that the black-box extractor outputs an accepting transcript together with $G(\mathbb{w})$. For falsifiability, we require that one can test whether $(\mathbb{x},\mathbb{w}) \notin \mathcal{R}_{\mathfrak{J}}$ when only given $(\mathbb{x},G(\mathbb{w}))$. This holds in our applications.

Definition 5 (Semi-Adaptive Black-Box G-Knowledge-Soundness). *Let $\mathfrak{J}$ be an R1CS instance with $n = n(\lambda)$ constraints. There exists a black-box expected (deterministic) PT extractor $\mathsf{Ext}_{\mathsf{ks}}$, such that for all non-uniform PPT $\mathcal{A}_1$, $\mathcal{D}$ and DPT $\mathcal{A}_2$, $\mathsf{Adv}^{\mathrm{bbks}}_{\mathsf{Pgen},G,\Pi,\mathsf{Ext}_{\mathsf{ks}},\mathcal{A}}(\lambda) := |\varepsilon_2(\lambda) - \varepsilon_1(\lambda)| \approx_\lambda 0$, where*

$$\varepsilon_1(\lambda) := \Pr\left[\mathcal{D}(\mathsf{lp},\mathsf{tr}) = 1 \middle| \begin{array}{l} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); \\ ((C,\mathbb{x}),\mathsf{st}) \leftarrow \mathcal{A}_1(\mathsf{lp},\mathcal{R}_{\mathfrak{J}}); (\mathsf{crs},\mathsf{td}) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},0); \\ \pi \leftarrow \mathcal{A}_2(\mathsf{st},\mathsf{crs}); \mathsf{tr} \leftarrow (C,\mathbb{x},\mathsf{crs},\pi) \end{array}\right] ,$$

$$\varepsilon_2(\lambda) := \Pr\left[\begin{array}{l} \mathcal{D}(\mathsf{lp},\mathsf{tr}) = 1 \wedge \\ \left(\begin{array}{c} \mathsf{V}(\mathsf{ck},\mathsf{tr}) = 1 \Rightarrow \\ (\mathbb{x},\mathbb{w}) \notin \mathcal{R}_{\mathfrak{J}} \end{array}\right) \end{array} \middle| \begin{array}{l} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); \\ ((C,\mathbb{x}),\mathsf{st}) \leftarrow \mathcal{A}_1(\mathsf{lp},\mathcal{R}_{\mathfrak{J}}); \\ (\mathsf{crs},\pi,G(\mathbb{w})) \leftarrow \mathsf{Ext}_{\mathsf{ks}}^{\mathcal{A}_2(\mathsf{st},\cdot)}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},C,\mathbb{x},\mathsf{st}); \\ \mathsf{tr} \leftarrow (C,\mathbb{x},\mathsf{crs},\pi) \end{array}\right] .$$

$\mathsf{Ext}_{\mathsf{ks}}$ is an oracle machine that makes an expected polynomial number of (adaptive or non-adaptive) queries. Before each query, $\mathsf{Ext}_{\mathsf{ks}}$ chooses $\varrho \in [1,n]$ and samples $(\mathsf{crs}^\varrho,\mathsf{td}^\varrho) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},\varrho)$. $\mathsf{Ext}_{\mathsf{ks}}$ then calls $\mathcal{A}_2(\mathsf{st},\mathsf{crs}^\varrho)$, obtaining some (possibly invalid) argument π^ϱ (st is not updated between $\mathcal{A}_2$ queries).

We allow $\mathsf{Ext}_{\mathsf{ks}}$ to use the same ϱ several times, but each time, $\mathsf{Ext}_{\mathsf{ks}}$ can use a different crs. In this case, π^ϱ depends on crs^ϱ and not only ϱ, but we will mostly ignore this detail. Let $\mathcal{Q}$ be the set of ϱ-s, actually used by $\mathsf{Ext}_{\mathsf{ks}}$. A C&P zk-SNARK is a *C&P SA-SNARK (semi-adaptive SNARK)* if it meets Definition 5.

Comparison to WEE. Compared to standard WEE [48], there are several differences. We can think of a semi-adaptive SNARG as a three-round protocol with a trusted setup, where the CRS is the verifier's second message. However, (1) the CRS is not public-coin, and (2) the CRS does not depend on the first message — it instead depends on the constraint number ϱ. Thus, our soundness notion and proof differ from the classical WEE ones. We use the name of black-box knowledge-soundness, although WEE might be more apt.

Comparison to [15]. In the context of (non-C&P) SNARGs for NP, Choudhuri et al. [15] define semi-adaptivity differently. Choudhuri et al. do not consider C&P arguments, but they allow for CRS reprogramming. In their semi-adaptivity game, the adversary first maliciously chooses the constraint ϱ, the CRS is programmed to use ϱ, and finally, the adversary outputs a statement and an argument. In our case, ϱ must stay hidden from the adversary; hence, we introduce the requirement of CRS-indistinguishability.

On G in G-knowledge-soundness. Since the lack of a trapdoor prevents one from efficiently computing $\mathbb{w}$ from $G(\mathbb{w})$, G-knowledge-soundness is a standard notion in many pairing-based schemes like Groth-Sahai. See [5,24] for further discussions. Since we work in the pairing-based setting, we set $G(s) := [sy]_1$ (we need y for compatibility with VCF). Involving $[\cdot]_1$ is a usual restriction in the pairing-based setting due to the hardness of the discrete logarithm.

A C&P SA-SNARK must satisfy one more requirement. $\mathsf{Ext}_{\mathsf{ks}}$ in Definition 5 can query $\mathcal{A}_2$ with CRSs corresponding to different constraints ϱ. The adversary's success is the difference between the probabilities of acceptance and extraction. In our case, it is crucial that if the adversary succeeds with a non-negligible probability, it does so for *any* $\varrho \in \mathcal{Q}$. Otherwise, the extractor might "miss" two inconsistent partial witnesses. We solve this by requiring CRS indistinguishability: CRSs for different ϱ are computationally indistinguishable. If that holds, the acceptance probability is roughly the same for different ϱ; hence, if the verifier accepts with a non-negligible probability, it does so for every ϱ. Crucially, the values extracted by the FSE somewhere-extractor when using different ϱ's do not have to be consistent; the reduction $\mathcal{B}_{\mathsf{fposb}}$ (see Section 6) handles this case.

Definition 6 (CRS-Indistinguishability). *For all λ, PPT $\mathcal{A}$, and $\varrho \in [1, n]$,* $\mathsf{Adv}^{\mathrm{crsind}}_{\mathsf{Pgen},\varrho,\Pi,\mathcal{A}}(\lambda) :=$

$$\Pr\left[\beta' = \beta \,\middle|\, \begin{array}{l} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); \beta \leftarrow\!\!\$\ \{0,1\}; \\ (\mathsf{crs}, \mathsf{td}) \leftarrow\!\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \beta \cdot \varrho); \beta' \leftarrow \mathcal{A}(\mathsf{lp}, \mathsf{crs}) \end{array}\right] \approx_\lambda \tfrac{1}{2} \ .$$

Special Soundness. We define a tailored special soundness [16] notion, semi-adaptive computational (k, G)-special soundness. Defining special soundness is a common step for interactive arguments but novel for non-interactive ones. We prove that any semi-adaptively computationally (n, G)-specially-sound and CRS-indistinguishable QA-SNARG Π is semi-adaptively black-box G-knowledge-sound. As typical in similar reductions, the knowledge-soundness extractor is only *expected* PPT. Later, we prove that the new zk-SNARK Punic

is semi-adaptively computationally (n, G)-specially-sound under three (strict) PPT computational assumptions.

Definition 7 (Semi-Adaptive Computational (k, G)-Special Soundness). *Fix $k \in \mathsf{poly}(\lambda)$. There exists a black-box PPT extractor $\mathsf{Ext}_{\mathsf{ss}}$, such that for any PPT adversary $\mathcal{A}_{\mathsf{ss}}$, $\mathsf{Adv}^{\mathrm{specsound}}_{\mathsf{Pgen},G,\Pi,k,\mathsf{Ext}_{\mathsf{ss}},\mathcal{A}_{\mathsf{ss}}}(\lambda) :=$*

$$\Pr\left[\begin{array}{l} \mathbf{tr} = (\mathsf{tr}^j)_{j=1}^k \wedge \\ \forall j \in [1,k]. \left(\begin{array}{c} \mathsf{tr}^j = (C, \mathbb{x}, \mathsf{crs}^j, \mathsf{td}^j, \pi^j) \\ \wedge\, (\mathsf{crs}^j, \mathsf{td}^j) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho^j) \\ \wedge\, \mathsf{V}(\mathsf{lp}, \mathsf{crs}^j, (C, \mathbb{x}), \pi^j) = 1 \end{array}\right) \\ \wedge\ \left(\forall i \neq j. \varrho^i \neq \varrho^j\right) \wedge\ (\mathbb{x}, \mathbb{w}) \notin \mathcal{R}_{\mathfrak{J}} \end{array} \middle| \begin{array}{l} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \\ \mathsf{lp} \leftarrow \mathsf{K}_{\mathsf{lp}}(\mathsf{p}); \\ \mathbf{tr} \leftarrow \mathcal{A}_{\mathsf{ss}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}); \\ G(\mathbb{w}) \leftarrow \mathsf{Ext}_{\mathsf{ss}}(\mathsf{lp}, \mathbf{tr}) \end{array}\right] \approx_\lambda 0\ .$$

Intuitively, Definition 7 states that if $\mathcal{A}_{\mathsf{ss}}$ produces an accepting admissible k-tuple $\mathbf{tr}$ (meaning that $\mathbf{tr}$ satisfies all conditions on the left-hand side), then one can — except with a negligible probability — black-box extract $G(\mathbb{w})$, such that $(\mathbb{x}, \mathbb{w}) \in \mathcal{R}_{\mathfrak{J}}$. The transcripts include trapdoors, needed in the special soundness proof of Punic. We assume that td^j contains ϱ^j.

The following result is related to yet different from classical reductions of WEE to special soundness. Note that $\mathcal{A}_{\mathsf{ss}}$ in Fig. 11 works in expected PPT. One can use Markov's inequality to make $\mathcal{A}_{\mathsf{ss}}$ to be strict PPT but with some loss in the probability. The latter technique is standard, and we will not elaborate on it. See Appendix D.3 for the proof of Theorem 1.

Theorem 1. *Let G be a permutation. If Π is semi-adaptively computationally (n, G)-special-sound and CRS-indistinguishable, then it is semi-adaptively black-box G-knowledge-sound for any family of instances $\mathfrak{J} = \mathfrak{J}(\lambda)$ with $n = n(\lambda)$ constraints. More precisely, there exists a black-box expected PPT extractor $\mathsf{Ext}_{\mathsf{ks}}$ and an expected PPT adversary $\mathcal{A}_{\mathsf{ss}}$, such that for any PPT $\mathsf{Ext}_{\mathsf{ss}}$ and $\mathcal{A}_{\mathsf{ks}} = (\mathcal{A}_1, \mathcal{A}_2)$, $\mathsf{Adv}^{\mathrm{bbks}}_{\mathsf{Pgen},G,\Pi,\mathsf{Ext}_{\mathsf{ks}},\mathcal{A}_{\mathsf{ks}}}(\lambda) \leq \mathsf{Adv}^{\mathrm{specsound}}_{\mathsf{Pgen},G,\Pi,n,\mathsf{Ext}_{\mathsf{ss}},\mathcal{A}_{\mathsf{ss}}}(\lambda)$.*

6 New C&P SA-SNARK Punic

Next, we propose a C&P SA-SNARK Punic for $\mathrm{R1CS}_{\mathsf{f}}$ by following ideas from [18,19,52]. We will use a new proof technique based on fully algebraic F-position-binding vector commitments and new security notions.

6.1 Intuition

We construct a C&P SA-SNARK Punic for $\mathrm{R1CS}_{\mathsf{f}}$ for a small constant f. Let $\mathsf{lp} = \mathrm{VCF.ck}$ and (x, y) be the VCF trapdoor key. The prover's statement is $([C(x)]_1, \mathbb{x})$, where $[C(x)]_1$ is a succinct VCF commitment to $\mathbb{z} = \left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right)$. Notably, (honest) crs is independent of the statement. Thus, crs can be created before the statement; we only prove soundness if the statement does not depend on crs.

The argument π includes three VCF commitments $([\mathsf{a}(x), \mathsf{c}(x)]_1$ and $[\mathsf{b}(x)]_2$ to $\boldsymbol{U}\mathbb{z}$, $\boldsymbol{W}\mathbb{z}$, and $\boldsymbol{V}\mathbb{z})$ and a group element $[\mathsf{h}(x)]_1$. Here, $\mathsf{h}(X) = (\mathsf{a}(X)\mathsf{b}(X) -$

$\mathsf{c}(X))/Z_{\mathbb{H}}(X)$. Intuitively, $[\mathsf{h}(x)]_1$ is the randomizer of the VCF commitment $[\mathsf{a}(x)\mathsf{b}(x) - \mathsf{c}(x)]_1$, Many non-universal zk-SNARKs, e.g. [25,56,31], have commitments $[\mathsf{a}(x), \mathsf{c}(x)]_1$ and $[\mathsf{b}(x)]_2$ and possibly the proof element $[\mathsf{h}(x)]_1$. Our novelty is using VCF, a *vector* commitment. Following [18,19], we prove that the commitments are correct (in particular, they commit to the correct public input) by using a BLS argument BLS.π that we add to Punic's argument.

The black-box extractor in our soundness proof extracts the local proofs corresponding to these three vector commitments. We follow [18,19] and add to the argument two FSE commitments $[\mathsf{d}(x)]_1$ and $[\mathsf{e}(x)]_2$ that allow us to black-box somewhere-extract one partial witness. For black-box extraction to succeed, the length of FSE commitments needs to be at least f group elements.

Soundness proof. Following the discussion of Section 5.2, we aim for Punic to be semi-adaptively $[\cdot]_1$-knowledge-sound—a different soundness notion than in [18,19]. Since we proved in Theorem 1 that this notion follows from special soundness, we will explain next how we prove special soundness. This helps to motivate the choice of primitives (VCF, FSE, and BLS).

In the honest case, $[C]_1$ is a VCF commitment to a statement-witness pair. We construct a special soundness extractor $\mathsf{Ext}_{\mathsf{ss}}$ (see Fig. 12). We also construct three reductions that work when the extractor $\mathsf{Ext}_{\mathsf{ss}}$ fails. These three reductions each call $\mathsf{Ext}_{\mathsf{ss}}$ to obtain a tuple of admissible transcripts $\mathbf{tr}$. Let $G(s) := [sy]_1$ to be G_1 from Section 4.2. Denote $G(\mathbb{p}^\varrho) := G(\mathbb{p}^\varrho(N(\varrho)))$. Each reduction loops over $\varrho \in [1, n]$. For each $\varrho \in [1, n]$, some of the reductions use FSE to black-box somewhere-extract $G(\mathbb{p}^\varrho) = G(\eta|_{N(\varrho)})$ together with $[\varphi|_{N(\varrho)}]_1$. Here, $\eta|_{N(\varrho)}$ is an assignment of variables from $N(\varrho)$, $[\varphi|_{N(\varrho)}]_1$ is a tuple of VCF local proofs for every coefficient in $N(\varrho)$, and $\mathsf{LVer}(\mathrm{VCF}_1.\mathsf{ck}, [C(x)]_1, k, G(\eta_k), [\varphi_k]_1) = 1$ for all $k \in N(\varrho)$. (We extract more values, but they are immaterial for this subsection.)

The first reduction $\mathcal{B}_{\mathsf{bls}}$ (see Fig. 13) is to the security of BLS. $\mathcal{B}_{\mathsf{bls}}$ guarantees three things: (1) the adversary uses the correct statement $\mathbb{x}$, (2) commitments like $[\mathsf{a}(x)]_1$ in the argument (see Fig. 5) are correctly formed, and (3) the extracted variables contain correctly computed local openings and local proofs of the vector commitment. Assuming that (1–3) holds, the second reduction $\mathcal{B}_{\mathsf{qal}}$ (see Fig. 15) handles the case when there exists a ϱ such that $\mathbb{p}^\varrho$ does not satisfy the ϱth coefficient. By the first two reductions, we obtain a guarantee for local consistency: for all ϱ, $(\mathbb{x}, \mathbb{p}^\varrho)$ locally satisfies the instance. The first two reductions are related to the reductions in [18,19], see Lemmas 5 and 7 for more.

The third reduction $\mathcal{B}_{\mathsf{fposb}}$ (see Fig. 14) handles the case when partial witnesses exist, but $\mathsf{Ext}_{\mathsf{ks}}$ fails to black-box extract a full witness satisfying all constraints. By Remark 1, then there must exist two indices $i \neq j$, such that:
(1) $\mathbb{p}^i$ satisfies the ith constraint and $\mathbb{p}^j$ satisfies the jth constraint.
(2) $\mathsf{Cons}(\mathbb{p}^i, \mathbb{p}^j) = \mathsf{false}$; that is, $(\exists k \in (N(i) \cap N(j)))\, \mathbb{p}^i(k) \neq \mathbb{p}^j(k)$.
Given all extracted $G(\mathbb{p}^\varrho)$-s, $\mathcal{B}_{\mathsf{fposb}}$ can efficiently recover i, j, k. $\mathcal{B}_{\mathsf{fposb}}$ returns the position k and two different local openings $\eta_k^i \neq \eta_k^j$ of $[C(x)]_1$ with local proofs φ_k^i and φ_k^j. Thus, $\mathcal{B}_{\mathsf{fposb}}$ breaks the $[\cdot]_1$-position-binding property.

Recall that FSE can black-box somewhere-extract group elements. Moreover, the extracted group elements must be linear maps of $\mathbb{z}$, that is, of the form $[\boldsymbol{M}]_\gamma \mathbb{z}$

for some public matrix $[\boldsymbol{M}]_\gamma$. Thus, the vector commitment scheme must be F-position-binding and fully-algebraic, which motivates the use of VCF.

In the ϱth iteration, we need to black-box extract η_k^ϱ and φ_k^ϱ for all $k \in N(\varrho)$. Since the length of an FSE commitment depends on the number of extracted values, we must limit the maximum number of such coefficients for the sake of efficiency. Thus, we can only handle $\mathrm{R1CS}_\mathfrak{f}$ for a small $\mathfrak{f}$.

We need protection against adversaries who make the verifier accept only for specific values of ϱ, which makes it impossible to construct $\mathbb{p}^\varrho$ for all ϱ. As explained in Section 5.2, it suffices to prove that Punic is CRS-indistinguishable.

See comparison with no-signaling commitments in Appendix A.2.

6.2 Description of Punic

Prerequisites. Punic uses VCF in $\mathbb{G}_1$ and $\mathbb{G}_2$ to commit. We use its local opening only in the soundness proof and not in the construction. Punic also uses FSE and BLS. Punic handles $\mathrm{R1CS}_\mathfrak{f}$, where $\mathfrak{f} \in \mathbb{N}$ is a small integer. Zk-SNARG(K)s for similarly restricted constraint systems are well-known; see, e.g., [23,41,57]. Using small $\mathfrak{f}$ only affects the efficiency: the restriction on $\mathfrak{f}$ is like to bounding the fan-in and fan-out in arithmetic circuits; it is easy to transform arithmetic circuits to circuits with bounded fan-in and fan-out efficiently.

Since we need to black-box extract the neighborhood of any given constraint, FSE has larger locality parameters than [18,19,52]. We set

$$q_1 = 2 + 2\mathfrak{f} \text{ and } q_2 := 2 \ . \tag{3}$$

We explain this choice in the soundness proof, see Appendix E. We use q_γ as the locality parameter for FSE_γ.

Description. In Fig. 5, we depict Punic for an $\mathrm{R1CS}_\mathfrak{f}$ instance $\mathfrak{J}$. The language parameter lp is the commitment key of VCF. For $\varrho \in [0, n]$ (in the honest case, $\varrho = 0$), Punic's CRS $\mathsf{crs} \leftarrow\!\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_\mathfrak{J}, \varrho)$ contains instance-dependent values $\mathrm{BLS.lp}$ and $\mathrm{BLS.crs}$ (BLS's language parameter and CRS). Furthermore, $\mathrm{BLS.lp} = [\boldsymbol{M}]_*$ contains as submatrices FSE commitment keys, together with commitment keys like ck_u required to locally open linear maps (see Lemma 4).

The FSE commitment keys are created as in Fig. 3 from ϱ-dependent extraction matrices $[\boldsymbol{E}_1^\varrho]_1$ and $[\boldsymbol{E}_2^\varrho]_2$. Here, $[\boldsymbol{E}_1^0]_1 = [\mathbf{0}_{q_1 \times (m+3)}]_1$ and $[\boldsymbol{E}_2^0]_2 = [\mathbf{0}_{q_2 \times (m+1)}]_2$. In the knowledge-soundness proof, we invoke $\mathsf{K}_{\mathsf{crs}}$ with a non-zero $\varrho \in [1, n]$. If $\varrho \neq 0$, then each row of $[\boldsymbol{E}_1^\varrho]_1$ / $[\boldsymbol{E}_2^\varrho]_2$ contains an extraction key used in the soundness proof to black-box extract local openings and local proofs. We describe the algorithm for creating $[\boldsymbol{E}_1^\varrho]_1$ / $[\boldsymbol{E}_2^\varrho]_2$ in Fig. 7. (We postpone it to Appendix E since the case $\varrho \neq 0$ is only used in the soundness proof.) Similarly, $[\boldsymbol{M}]_*$ is created by using the algorithm in Fig. 6. In Figs. 6 and 7, the first row (small, blue font) denotes the elements of the vector that the matrices will be multiplied with. "Empty" entries mean zeros. We explain the construction of these matrices in Section 7.1.

$\mathsf{K}_{\mathsf{lp}}(\mathsf{p})$: // $\mathrm{VCF}_2.\mathsf{ck} = \mathrm{VCF}_1.\mathsf{ck}$, $\mathrm{VCF}_2.\mathsf{td} = \mathrm{VCF}_1.\mathsf{td}$ by design
$(\mathrm{VCF}_1.\mathsf{ck}, \mathrm{VCF}_1.\mathsf{td}) \leftarrow \mathrm{VCF}_1.\mathsf{K}_{\mathsf{ck}}(\mathsf{p}, n)$; // $\mathrm{VCF}_1.\mathsf{td}$ contains x, y
return $\mathsf{lp} \leftarrow \mathrm{VCF}_1.\mathsf{ck} = \mathrm{VCF}_2.\mathsf{ck}$;

$\mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho)$: // n is implicit in $\mathsf{p}, \mathcal{R}_{\mathfrak{J}}$; honest case: $\varrho = 0$
$([\boldsymbol{E}_1^{\varrho}]_1, [\boldsymbol{E}_2^{\varrho}]_2) \leftarrow \mathsf{CreateE}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho)$; // FSE extraction matrices
$(\mathrm{FSE}_1.\mathsf{ck}, \mathrm{FSE}_1.\mathsf{td}) \leftarrow \mathrm{FSE}_1.\mathsf{K}_{\mathsf{ck}}(\mathsf{p}, m+3, q_1, [\boldsymbol{E}_1^{\varrho}]_1)$; // As in Fig. 3
$(\mathrm{FSE}_2.\mathsf{ck}, \mathrm{FSE}_2.\mathsf{td}) \leftarrow \mathrm{FSE}_2.\mathsf{K}_{\mathsf{ck}}(\mathsf{p}, m+1, q_2, [\boldsymbol{E}_2^{\varrho}]_2)$;
$\mathrm{BLS}.\mathsf{lp} = [\boldsymbol{M}]_* \leftarrow \mathsf{CreateM}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \mathrm{FSE}_1.\mathsf{ck}, \mathrm{FSE}_2.\mathsf{ck})$;
$(\mathrm{BLS}.\mathsf{crs}, \mathrm{BLS}.\mathsf{td}) \leftarrow \mathrm{BLS}.\mathsf{K}_{\mathsf{crs}}(\mathsf{p}, \mathrm{BLS}.\mathsf{lp})$;
$\mathsf{crs} \leftarrow (\mathrm{BLS}.\mathsf{lp}, \mathrm{BLS}.\mathsf{crs})$;
$\mathsf{ek} \leftarrow (\mathrm{FSE}_1.\mathsf{ek}, \mathrm{FSE}_2.\mathsf{ek})$; $\mathsf{td} \leftarrow (\mathrm{BLS}.\mathsf{td}, \mathsf{ek})$;
return $(\mathsf{crs}, \mathsf{td})$;

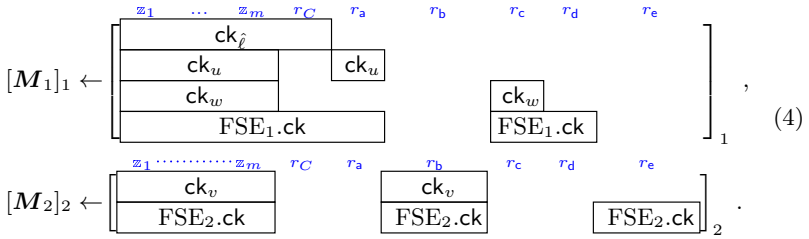
$\mathsf{P}(\mathsf{lp}, \mathsf{crs}, ([C(x)]_1, \mathbb{x} \in \mathbb{F}^{m_{\mathbb{x}}}), (r_C, \mathbb{w} \in \mathbb{F}^{m-m_{\mathbb{x}}}))$:
// $\mathbb{z} = \left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right)$; $[C(x)]_1 \leftarrow \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \mathbb{z}, r_C)$
1. $r_{\mathsf{a}}, r_{\mathsf{b}}, r_{\mathsf{c}}, r_{\mathsf{d}}, r_{\mathsf{e}} \leftarrow\$ \mathbb{F}$;
2. $[\mathsf{a}(x)]_1 \leftarrow \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \boldsymbol{U}\mathbb{z}; r_{\mathsf{a}})$;
3. $[\mathsf{b}(x)]_2 \leftarrow \mathrm{VCF}_2.\mathsf{Com}(\mathrm{VCF}_2.\mathsf{ck}, \boldsymbol{V}\mathbb{z}; r_{\mathsf{b}})$;
4. $[\mathsf{c}(x)]_1 \leftarrow \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \boldsymbol{W}\mathbb{z}; r_{\mathsf{c}})$;
5. $\mathsf{h}(X) \leftarrow (\mathsf{a}(X)\mathsf{b}(X) - \mathsf{c}(X))/Z_{\mathbb{H}}(X)$; // $[\mathsf{h}(x)]_1 \leftarrow \sum_{i=0}^{n-2} h_i[x^i]_1$
6. $[\mathsf{d}(x)]_1 \leftarrow \mathrm{FSE}_1.\mathsf{Com}(\mathrm{FSE}_1.\mathsf{ck}, (\mathbb{z}^{\intercal}, r_C, r_{\mathsf{a}}, r_{\mathsf{c}})^{\intercal}; r_{\mathsf{d}})$;
7. $[\mathsf{e}(x)]_2 \leftarrow \mathrm{FSE}_2.\mathsf{Com}(\mathrm{FSE}_2.\mathsf{ck}, \left(\begin{smallmatrix}\mathbb{z}\\ r_{\mathsf{b}}\end{smallmatrix}\right); r_{\mathsf{e}})$;
8. $[C^*(x)]_1 \leftarrow [C(x)]_1 - \sum_{i=1}^{m_{\mathbb{x}}} \mathbb{x}_i[\ell_i(x)]_1$;
9. $\mathrm{BLS}.\mathbb{x} \leftarrow ([C^*(x), \mathsf{a}(x), \mathsf{c}(x), \mathsf{d}(x)]_1, [\mathsf{b}(x), \mathsf{e}(x)]_2)^{\intercal}$;
10. $\mathrm{BLS}.\pi \leftarrow \mathrm{BLS}.\mathsf{P}(\mathrm{BLS}.\mathsf{lp}, \mathrm{BLS}.\mathsf{crs}, \mathrm{BLS}.\mathbb{x}, (\mathbb{z}, r_C, r_{\mathsf{a}}, r_{\mathsf{b}}, r_{\mathsf{c}}, r_{\mathsf{d}}, r_{\mathsf{e}}))$;
11. $\pi \leftarrow ([\mathsf{a}(x), \mathsf{c}(x), \mathsf{d}(x), \mathsf{h}(x)]_1, [\mathsf{b}(x), \mathsf{e}(x)]_2, \mathrm{BLS}.\pi)$;

$\mathsf{V}(\mathsf{lp}, \mathsf{crs}, ([C(x)]_1, \mathbb{x} \in \mathbb{F}^{m_{\mathbb{x}}}), \pi)$: Parse π as in 11;
1. $[C^*(x)]_1 \leftarrow [C(x)]_1 - \sum_{i=1}^{m_{\mathbb{x}}} \mathbb{x}_i[\ell_i(x)]_1$;
2. $\mathrm{BLS}.\mathbb{x} \leftarrow ([C^*(x), \mathsf{a}(x), \mathsf{c}(x), \mathsf{d}(x)]_1, [\mathsf{b}(x), \mathsf{e}(x)]_2)^{\intercal}$;
3. check $\mathrm{BLS}.\mathsf{V}(\mathrm{BLS}.\mathsf{lp}, \mathrm{BLS}.\mathsf{crs}, \mathrm{BLS}.\mathbb{x}, \mathrm{BLS}.\pi) = 1$;
4. check $[\mathsf{a}(x)]_1 \bullet [\mathsf{b}(x)]_2 - [\mathsf{c}(x)]_1 \bullet [1]_2 = [\mathsf{h}(x)]_1 \bullet [Z_{\mathbb{H}}(x)]_2$;

$\mathsf{Sim}(\mathsf{lp}, \mathsf{crs}, \mathsf{td} = (\mathrm{BLS}.\mathsf{td}, \mathsf{ek}), ([C(x)]_1, \mathbb{x} \in \mathbb{F}^{m_{\mathbb{x}}}))$:
1. $r_{\mathsf{a}}, r_{\mathsf{b}}, r_{\mathsf{c}}, r_{\mathsf{d}}, r_{\mathsf{e}} \leftarrow\$ \mathbb{F}$;
2. $[\mathsf{a}(x)]_1 \leftarrow \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \mathbf{0}; r_{\mathsf{a}})$; // $= r_{\mathsf{a}}[Z_{\mathbb{H}}(x)]_1$
3. $[\mathsf{b}(x)]_2 \leftarrow \mathrm{VCF}_2.\mathsf{Com}(\mathrm{VCF}_2.\mathsf{ck}, \mathbf{0}; r_{\mathsf{b}})$; // $= r_{\mathsf{b}}[Z_{\mathbb{H}}(x)]_2$
4. $[\mathsf{c}(x)]_1 \leftarrow \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \mathbf{0}; r_{\mathsf{c}})$; // $= r_{\mathsf{c}}[Z_{\mathbb{H}}(x)]_1$
5. $[\mathsf{h}(x)]_1 \leftarrow r_{\mathsf{a}} r_{\mathsf{b}}[Z_{\mathbb{H}}(x)]_1 - r_{\mathsf{c}}[1]_1$;
6. $[\mathsf{d}(x)]_1 \leftarrow \mathrm{FSE}_1.\mathsf{Com}(\mathrm{FSE}_1.\mathsf{ck}, \mathbf{0}_{m+3}; r_{\mathsf{d}})$;
7. $[\mathsf{e}(x)]_2 \leftarrow \mathrm{FSE}_2.\mathsf{Com}(\mathrm{FSE}_2.\mathsf{ck}, \mathbf{0}_{m+1}; r_{\mathsf{e}})$;
8. $[C^*(x)]_1 \leftarrow [C(x)]_1 - \sum_{i=1}^{m_{\mathbb{x}}} \mathbb{x}_i[\ell_i(x)]_1$;
9. $\mathrm{BLS}.\mathbb{x} \leftarrow ([C^*(x), \mathsf{a}(x), \mathsf{c}(x), \mathsf{d}(x)]_1, [\mathsf{b}(x), \mathsf{e}(x)]_2)^{\intercal}$;
10. $\mathrm{BLS}.\pi \leftarrow \mathrm{BLS}.\mathsf{Sim}(\mathrm{BLS}.\mathsf{lp}, \mathrm{BLS}.\mathsf{crs}, \mathrm{BLS}.\mathsf{td}, \mathrm{BLS}.\mathbb{x})$;
11. $\pi \leftarrow ([\mathsf{a}(x), \mathsf{c}(x), \mathsf{d}(x), \mathsf{h}(x)]_1, [\mathsf{b}(x), \mathsf{e}(x)]_2, \mathrm{BLS}.\pi)$;

Fig. 5. New semi-adaptively black-box $[\cdot]_1$-knowledge-sound C&P SA-SNARK Punic.

1. let $\hat{\ell}_j(X) = \ell_j(X)$ for $m_{\mathbb{x}} < j \le n$ and $\hat{\ell}_j(X) = 0$ for $j \le m_{\mathbb{x}}$ or $j > n$;
   // W.l.o.g., we assume $m \ge n$
2. let $u_j(X), v_j(X), w_j(X)$ interpolate column vectors $\boldsymbol{U}^{(j)}, \boldsymbol{V}^{(j)}, \boldsymbol{W}^{(j)}$;
3. $[\mathsf{ck}_{\hat{\ell}}]_1 \leftarrow [\hat{\ell}_1(x), \ldots, \hat{\ell}_m(x), Z_{\mathbb{H}}(x)]_1$; $[\mathsf{ck}_u]_1 \leftarrow [u_1(x), \ldots, u_m(x), Z_{\mathbb{H}}(x)]_1$;
4. $[\mathsf{ck}_v]_2 \leftarrow [v_1(x), \ldots, v_m(x), Z_{\mathbb{H}}(x)]_2$; $[\mathsf{ck}_w]_1 \leftarrow [w_1(x), \ldots, v_m(x), Z_{\mathbb{H}}(x)]_1$;
5. let $n_1 := q_1 + 4$, $n_2 := q_2 + 2$;
6. return $[\boldsymbol{M}]_* \in \mathbb{G}_1^{n_1 \times (m+6)} \times \mathbb{G}_2^{n_2 \times (m+6)}$, where



(4)

Fig. 6. Algorithm $\mathsf{CreateM}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \mathrm{FSE}_1.\mathsf{ck}, \mathrm{FSE}_2.\mathsf{ck})$.

Efficiency. Clearly, $\mathrm{FSE}_1.\mathsf{ck} \in \mathbb{G}_1^{(q_1+1)\times(m+3)}$ and $\mathrm{FSE}_2.\mathsf{ck} \in \mathbb{G}_2^{(q_2+1)\times(m+2)}$. Using $\hat{\ell}_i(X)$ instead of $\ell_i(X)$ helps us to prove efficiently that the prover used the correct $\mathrm{R1CS}_\mathfrak{f}$ statement $(\mathbb{z}_1, \ldots, \mathbb{z}_{m_{\mathbb{x}}})^\mathsf{T} = \mathbb{x}$. Assuming we have an instance of $\mathrm{R1CS}_\mathfrak{f}$ for $\mathfrak{f} = o(|\mathbb{w}|)$, the Punic argument π is succinct, consisting of $7 + 2\mathfrak{f}$ elements of $\mathbb{G}_1$ and 5 elements of $\mathbb{G}_2$. Choosing a larger $\mathfrak{f}$ potentially decreases the number of constraints, while a smaller $\mathfrak{f}$ decreases the argument size.

SSP. In Appendix B.3, we note that Punic can be simplified significantly by targeting SSP [17] instead of R1CS [25].

7 Security of Punic

We postpone the following two proofs to Appendices E.1 and E.2.

Theorem 2. *(1) Punic is perfectly complete. (2) If* VCF_1 *and* VCF_2 *are perfectly zero-knowledge,* BLS *is perfectly zero-knowledge, and* FSE_1 *and* FSE_2 *are almost everywhere perfectly-hiding then Punic is perfectly zero-knowledge.*

Theorem 3. *Let* m, q_1, *and* q_2 *be as above. If* FSE_γ *is function-set hiding for* $\gamma \in \{1, 2\}$, *then Punic is CRS-indistinguishable. More precisely, there exist PPT* $\mathcal{B}_1$ *and* $\mathcal{B}_2$, *such that for every PPT* $\mathcal{A}$ *and* ϱ, $\mathsf{Adv}^{\mathrm{crsind}}_{\mathsf{Pgen},\varrho,\mathrm{Punic},\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{Pgen},\mathrm{FSE}_1,m+3,q_1,\mathcal{B}_1}(\lambda) + \mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{Pgen},\mathrm{FSE}_2,m+1,q_2,\mathcal{B}_2}(\lambda)$.

7.1 Semi-Adaptive Computational (n, G)-Special-Soundness

On $\boldsymbol{M}_\gamma$. For $\mathrm{BLS}.\mathsf{lp} = [\boldsymbol{M}]_*$ defined as in Eq. (4), we use BLS to show that

$$\mathrm{BLS}.\mathbb{x} := ([C^*(x), \mathsf{a}(x), \mathsf{c}(x), \mathsf{d}(x)]_1, [\mathsf{b}(x), \mathsf{e}(x)]_2)^\mathsf{T} \in \mathrm{colspace}\begin{pmatrix} [\boldsymbol{M}_1]_1 \\ [\boldsymbol{M}_2]_2 \end{pmatrix} . \quad (5)$$

1. if $\varrho = 0$ then $\boldsymbol{E}_1^\varrho \leftarrow \mathbf{0}_{q_1\times(m+3)}$; $\boldsymbol{E}_2^\varrho \leftarrow \mathbf{0}_{q_2\times(m+1)}$; return $([\boldsymbol{E}_1^\varrho]_1, [\boldsymbol{E}_2^\varrho]_2)$;
2. $(\beta_1, \ldots, \beta_{\mathfrak{f}}) \leftarrow N(\varrho)$; // Duplicate some entries when $|N(\varrho)| < \mathfrak{f}$
3. let $u_j(X)$, $v_j(X)$, $w_j(X)$ interpolate column vectors $\boldsymbol{U}^{(j)}$, $\boldsymbol{V}^{(j)}$, $\boldsymbol{W}^{(j)}$;
4. as in Lemma 1, let $[\mathsf{ck}_{\ell,\beta_i}]_1 \leftarrow [Q_{\ell_1,\beta_i}(x)\| \ldots \|Q_{\ell_m,\beta_i}(x)\|Q_{Z_{\mathbb{H}},\beta_i}(x)]_1^\intercal$.
5. as in Lemma 4, let $[\mathsf{ck}_u^\varrho]_1 \leftarrow [Q_{u_1,\varrho}(x)\| \ldots \|Q_{u_m,\varrho}(x)\|Q_{Z_{\mathbb{H}},\varrho}(x)]_1^\intercal$; similarly, define $[\mathsf{ck}_v^\varrho]_2$, $[\mathsf{ck}_w^\varrho]_1$;
6. return $([\boldsymbol{E}_1^\varrho]_1, [\boldsymbol{E}_2^\varrho]_2)$, $\boldsymbol{E}_1^\varrho \in \mathbb{F}^{q_1\times(m+3)}$, $\boldsymbol{E}_2^\varrho \in \mathbb{F}^{q_2\times(m+1)}$, where

$$[\boldsymbol{E}_1^\varrho]_1 \leftarrow \left[\begin{array}{c|c|c|c} \multicolumn{2}{c|}{\boldsymbol{e}_{\beta_1}^\intercal y} & r_C & r_{\mathsf{a}} \quad r_{\mathsf{c}} \\ \multicolumn{2}{c|}{\cdots\cdots} & & \\ \multicolumn{2}{c|}{\boldsymbol{e}_{\beta_{\mathfrak{f}}}^\intercal y} & & \\ \multicolumn{2}{c|}{\mathsf{ck}_{\ell,\beta_1}} & & \\ \multicolumn{2}{c|}{\cdots\cdots} & & \\ \multicolumn{2}{c|}{\mathsf{ck}_{\ell,\beta_{\mathfrak{f}}}} & & \\ \mathsf{ck}_u^\varrho & & \mathsf{ck}_u^\varrho & \\ \mathsf{ck}_w^\varrho & & & \mathsf{ck}_w^\varrho \end{array}\right]_1, \qquad [\boldsymbol{E}_2^\varrho]_2 \leftarrow \left[\begin{array}{c|c|c} \multicolumn{2}{c|}{\boldsymbol{V}_\varrho^\intercal y} & r_{\mathsf{b}} \\ \multicolumn{2}{c|}{\mathsf{ck}_v^\varrho} & \end{array}\right]_2 . \tag{7}$$

Fig. 7. Algorithm $\mathsf{CreateE}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho)$, where $\varrho \in [0, n]$.

Eq. (5) holds iff there exists $\mathrm{BLS}.\mathbb{w} = (\mathbb{z} = \left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right), r_C, r_{\mathsf{a}}, r_{\mathsf{b}}, r_{\mathsf{c}}, r_{\mathsf{d}}, r_{\mathsf{e}})$, such that (here, $[C(x)]_1$ follows from $[C^*(x)]_1$)

$$\begin{aligned}
[C^*(x)]_1 &= \mathrm{VCF}_1.\mathsf{ck} \cdot \left(\begin{smallmatrix}\mathbf{0}\\ \mathbb{w}\\ r_C\end{smallmatrix}\right) , \\
[C(x)]_1 &= \mathrm{VCF}_1.\mathsf{ck} \cdot \left(\begin{smallmatrix}\mathbb{z}\\ r_C\end{smallmatrix}\right) = \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \mathbb{z}, r_C) , \\
[\mathsf{a}(x)]_1 &= \mathrm{VCF}_1.\mathsf{ck} \cdot \left(\begin{smallmatrix}\boldsymbol{U}\mathbb{z}\\ r_{\mathsf{a}}\end{smallmatrix}\right) = \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \boldsymbol{U}\mathbb{z}; r_{\mathsf{a}}) , \\
[\mathsf{c}(x)]_1 &= \mathrm{VCF}_1.\mathsf{ck} \cdot \left(\begin{smallmatrix}\boldsymbol{W}\mathbb{z}\\ r_{\mathsf{c}}\end{smallmatrix}\right) = \mathrm{VCF}_1.\mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \boldsymbol{W}\mathbb{z}; r_{\mathsf{c}}) , \\
[\mathsf{d}(x)]_1 &= \mathrm{FSE}_1.\mathsf{ck} \cdot (\mathbb{z}^\intercal, r_C, r_{\mathsf{a}}, r_{\mathsf{c}}, r_{\mathsf{d}})^\intercal \\
&= \mathrm{FSE}_1.\mathsf{Com}(\mathrm{FSE}_1.\mathsf{ck}, (\mathbb{z}^\intercal, r_C, r_{\mathsf{a}}, r_{\mathsf{c}})^\intercal; r_{\mathsf{d}}) , \\
[\mathsf{b}(x)]_2 &= \mathrm{VCF}_2.\mathsf{ck} \cdot \left(\begin{smallmatrix}\boldsymbol{V}\mathbb{z}\\ r_{\mathsf{b}}\end{smallmatrix}\right) = \mathrm{VCF}_2.\mathsf{Com}(\mathrm{VCF}_2.\mathsf{ck}, \boldsymbol{V}\mathbb{z}; r_{\mathsf{b}}) , \\
[\mathsf{e}(x)]_2 &= \mathrm{FSE}_2.\mathsf{ck} \cdot (\mathbb{z}^\intercal, r_{\mathsf{b}}, r_{\mathsf{e}})^\intercal = \mathrm{FSE}_2.\mathsf{Com}(\mathrm{FSE}_2.\mathsf{ck}, \left(\begin{smallmatrix}\mathbb{z}\\ r_{\mathsf{b}}\end{smallmatrix}\right); r_{\mathsf{e}}) .
\end{aligned} \tag{6}$$

By Fact 2, for BLS to be strongly sound, we need the distribution of $[\boldsymbol{M}]_*$ to be witness-sampleable; this is clearly the case. We also need that $\mathrm{rank}\left(\begin{smallmatrix}\boldsymbol{M}_1\\ \boldsymbol{M}_2\end{smallmatrix}\right) < n_1 + n_2$. This is fine since $\mathrm{BLS}.\mathbb{w}$ *always* exists when $n_1 + n_2 = \mathrm{rank}\left(\begin{smallmatrix}\boldsymbol{M}_1\\ \boldsymbol{M}_2\end{smallmatrix}\right)$.

On $\boldsymbol{E}_\gamma$. Assume $\varrho \neq 0$, $(\mathsf{crs}^\varrho, \mathsf{td}^\varrho) \leftarrow\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho)$, and that P computes the argument π^ϱ honestly by using an ϱ-dependent full witness $\mathbb{w}^\varrho$ and randomizers like r_{a}^ϱ. (BLS will guarantee the latter.) Then, $\mathrm{FSE}_1.\mathsf{swExt}(\mathrm{FSE}_1.\mathsf{ek}, [\mathsf{d}(x)]_1)$ and

$\mathrm{FSE}_2.\mathsf{swExt}(\mathrm{FSE}_2.\mathsf{ek}, [\mathsf{e}(x)]_2)$ output

$$\begin{pmatrix} G(\boldsymbol{\eta}^\varrho) \\ \left[\begin{smallmatrix} \boldsymbol{\varphi}^\varrho \\ \varphi^\varrho_{\mathsf{a}} \\ \varphi^\varrho_{\mathsf{c}} \end{smallmatrix}\right]_1 \end{pmatrix} \leftarrow [\boldsymbol{E}_1^\varrho]_1 \cdot \begin{pmatrix} \mathbb{z}^\varrho \\ r^\varrho_C \\ r^\varrho_{\mathsf{a}} \\ r^\varrho_{\mathsf{c}} \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} G_2(\eta^\varrho_{\mathsf{b}}) \\ [\varphi^\varrho_{\mathsf{b}}]_2 \end{pmatrix} \leftarrow [\boldsymbol{E}_2^\varrho]_2 \cdot \begin{pmatrix} \mathbb{z}^\varrho \\ r^\varrho_{\mathsf{b}} \end{pmatrix} , \tag{8}$$

where $G_2(X) := [Xy]_2$. From Lemma 1, the security of BLS, and Eq. (7) it follows $G(\boldsymbol{\eta}^\varrho|_{N(\varrho)}) = G(\mathbb{p}^\varrho|_{N(\varrho)})$ is a tuple of local openings and $\boldsymbol{\varphi}^\varrho = (\varphi_j^\varrho)|_{N(\varrho)}$ is the corresponding tuple of local proofs, with

$$(G(\eta_j^\varrho), [\varphi_j^\varrho]_1) = \mathrm{VCF}_1.\mathsf{LOpen}(\mathsf{ck}, [C(x)]_1, j, D = (\mathbb{z}^\varrho, r^\varrho))$$

for some $\mathbb{z}^\varrho$ and r^ϱ. (Recall that dependency from y is required to construct a reduction to $[\cdot]_1$-position-binding.) Define $G(\eta^\varrho_{\mathsf{a}}) \leftarrow \sum_{j\in N(\varrho)} U_{\varrho j} G(\eta_j^\varrho)$ and $G(\eta^\varrho_{\mathsf{c}}) \leftarrow \sum_{j\in N(\varrho)} W_{\varrho j} G(\eta_j^\varrho)$. Eqs. (7) and (8) and Lemma 4 imply that

$$(G(\eta^\varrho_{\mathsf{a}}), [\varphi^\varrho_{\mathsf{a}}]_1) = \mathrm{VCF}_1.\mathsf{LOpen}(\mathsf{ck}, [\mathsf{a}(x)]_1, j, D = (\boldsymbol{U}\mathbb{z}^\varrho, r^\varrho_{\mathsf{a}}))$$

and $(G(\eta^\varrho_{\mathsf{c}}), [\varphi^\varrho_{\mathsf{c}}]_1) = \mathrm{VCF}_1.\mathsf{LOpen}(\mathsf{ck}, [\mathsf{c}(x)]_1, j, D = (\boldsymbol{W}\mathbb{z}^\varrho, r^\varrho_{\mathsf{c}}))$. Note that we black-box extract $[\varphi^\varrho_{\mathsf{b}}]_2$ by using FSE.

For $\varrho \in [0, n]$, let $\mathcal{D}^\varrho_{\mathrm{par}}$ be the distribution of $[\boldsymbol{M}]_*$ in Eq. (4). We postpone the special soundness proof to Appendix E.3.

Theorem 4. *Let n be the number of R1CS$_{\mathsf{f}}$ constraints. Assume FSE_γ is somewhere $[\cdot]_\gamma$-extractable for $\gamma \in \{1, 2\}$, BLS is quasi-adaptively strongly sound for $\mathcal{D}^\varrho_{\mathrm{par}}$ where $\varrho \in [1, n]$, VCF_1 is $[\cdot]_1$-position-binding, and n-QALINRES holds. Then, Punic is semi-adaptively computationally (n, G)-special-sound. More precisely, there exist an expected PPT $\mathsf{Ext}_{\mathsf{ss}}$ and PPT $\mathcal{B}_{\mathsf{fposb}}$, $\mathcal{B}_{\mathsf{qal}}$, and $\mathcal{B}^\varrho_{\mathsf{bls}}$ for $\varrho \in [1, n]$, such that for any PPT $\mathcal{A}_{\mathsf{ss}}$,*

$$\mathsf{Adv}^{\mathrm{specsound}}_{\mathsf{Pgen},G,\mathrm{Punic},n,\mathsf{Ext}_{\mathsf{ss}},\mathcal{A}_{\mathsf{ss}}}(\lambda) \le \sum_{\varrho=1}^{n} \mathsf{Adv}^{\mathrm{strsound}}_{\mathsf{Pgen},\mathcal{D}^\varrho_{\mathrm{par}},\mathrm{BLS},\mathcal{B}^\varrho_{\mathsf{bls}}}(\lambda) + \mathsf{Adv}^{\mathrm{fposb}}_{\mathsf{Pgen},[\cdot]_1,n,\mathrm{VCF}_1,\mathcal{B}_{\mathsf{fposb}}}(\lambda) + \mathsf{Adv}^{\mathrm{qalinres}}_{\mathsf{Pgen},n,\mathcal{B}_{\mathsf{qal}}}(\lambda) \ .$$

Acknowledgment and History. The author became aware of the error in FANA in December 2021; the error was (partially) caused by the fact that he was severely sick when submitting [52] and its camera-ready version. We thank Daniel Wichs and anonymous reviewers for helpful comments.

References

1. Abdolmaleki, B., Baghery, K., Lipmaa, H., Zajac, M.: A subversion-resistant SNARK. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part III. LNCS, vol. 10626, pp. 3–33. Springer, Heidelberg (Dec 2017). https://doi.org/10.1007/978-3-319-70700-6_1
2. Abdolmaleki, B., Lipmaa, H., Siim, J., Zajac, M.: On QA-NIZK in the BPK model. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 590–620. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45374-9_20

3. Abdolmaleki, B., Lipmaa, H., Siim, J., Zajac, M.: On Subversion-Resistant SNARKs. J. Cryptology **34**(3), 1–42 (2021). https://doi.org/10.1007/s00145-021-09379-y
4. Agrawal, S., Dodis, Y., Vaikuntanathan, V., Wichs, D.: On continual leakage of discrete log representations. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part II. LNCS, vol. 8270, pp. 401–420. Springer, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-42045-0_21
5. Belenkiy, M., Chase, M., Kohlweiss, M., Lysyanskaya, A.: P-signatures and noninteractive anonymous credentials. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 356–374. Springer, Heidelberg (Mar 2008). https://doi.org/10.1007/978-3-540-78524-8_20
6. Bellare, M., Fuchsbauer, G., Scafuro, A.: NIZKs with an untrusted CRS: Security in the face of parameter subversion. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 777–804. Springer, Heidelberg (Dec 2016). https://doi.org/10.1007/978-3-662-53890-6_26
7. Ben-Sasson, E., Chiesa, A., Genkin, D., Tromer, E.: On the concrete efficiency of probabilistically-checkable proofs. In: Boneh, D., Roughgarden, T., Feigenbaum, J. (eds.) STOC 2013. pp. 585–594. ACM Press, Palo Alto, CA, USA (Jun 1–4, 2013). https://doi.org/10.1145/2488608.2488681
8. Ben-Sasson, E., Chiesa, A., Spooner, N.: Interactive oracle proofs. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part II. LNCS, vol. 9986, pp. 31–60. Springer, Heidelberg (Oct / Nov 2016). https://doi.org/10.1007/978-3-662-53644-5_2
9. Boneh, D., Boyen, X.: Short signatures without random oracles and the SDH assumption in bilinear groups. Journal of Cryptology **21**(2), 149–177 (Apr 2008). https://doi.org/10.1007/s00145-007-9005-7
10. Camenisch, J., Dubovitskaya, M., Haralambiev, K., Kohlweiss, M.: Composable and modular anonymous credentials: Definitions and practical constructions. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part II. LNCS, vol. 9453, pp. 262–288. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48800-3_11
11. Campanelli, M., Faonio, A., Fiore, D., Querol, A., Rodríguez, H.: Lunar: A toolbox for more efficient universal and updatable zkSNARKs and commit-and-prove extensions. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part III. LNCS, vol. 13092, pp. 3–33. Springer, Heidelberg (Dec 2021). https://doi.org/10.1007/978-3-030-92078-4_1
12. Campanelli, M., Ganesh, C., Khoshakhlagh, H., Siim, J.: Impossibilities in Succinct Arguments: Black-box Extraction and More. In: Duquesne, S., Feo, L.D., Mrabet, N.E. (eds.) AFRICACRYPT 2023. LNCS, vol. ?, pp. ?–? Springer, Cham, Sousse, Tunisia (Jul 19–21, 2023)
13. Catalano, D., Fiore, D.: Vector commitments and their applications. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 55–72. Springer, Heidelberg (Feb / Mar 2013). https://doi.org/10.1007/978-3-642-36362-7_5
14. Chiesa, A., Hu, Y., Maller, M., Mishra, P., Vesely, P., Ward, N.P.: Marlin: Preprocessing zkSNARKs with universal and updatable SRS. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part I. LNCS, vol. 12105, pp. 738–768. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45721-1_26
15. Choudhuri, A.R., Jain, A., Jin, Z.: SNARGs for $\mathcal{P}$ from LWE. In: FOCS 2021. pp. 68–79. IEEE, IEEE Computer Society Press, Denver, Colorado, USA (Feb, 7–10 2021)

16. Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Desmedt, Y. (ed.) CRYPTO'94. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (Aug 1994). https://doi.org/10.1007/3-540-48658-5_19
17. Danezis, G., Fournet, C., Groth, J., Kohlweiss, M.: Square span programs with applications to succinct NIZK arguments. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014, Part I. LNCS, vol. 8873, pp. 532–550. Springer, Heidelberg (Dec 2014). https://doi.org/10.1007/978-3-662-45611-8_28
18. Daza, V., González, A., Pindado, Z., Ràfols, C., Silva, J.: Shorter quadratic QA-NIZK proofs. In: Lin, D., Sako, K. (eds.) PKC 2019, Part I. LNCS, vol. 11442, pp. 314–343. Springer, Heidelberg (Apr 2019). https://doi.org/10.1007/978-3-030-17253-4_11
19. Fauzi, P., Lipmaa, H., Pindado, Z., Siim, J.: Somewhere statistically binding commitment schemes with applications. In: Borisov, N., Díaz, C. (eds.) FC 2021, Part I. LNCS, vol. 12674, pp. 436–456. Springer, Heidelberg (Mar 2021). https://doi.org/10.1007/978-3-662-64322-8_21
20. Feige, U., Lapidot, D., Shamir, A.: Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In: 31st FOCS. pp. 308–317. IEEE Computer Society Press (Oct 1990). https://doi.org/10.1109/FSCS.1990.89549
21. Fuchsbauer, G.: Subversion-zero-knowledge SNARKs. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part I. LNCS, vol. 10769, pp. 315–347. Springer, Heidelberg (Mar 2018). https://doi.org/10.1007/978-3-319-76578-5_11
22. Fuchsbauer, G., Kiltz, E., Loss, J.: The algebraic group model and its applications. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part II. LNCS, vol. 10992, pp. 33–62. Springer, Heidelberg (Aug 2018). https://doi.org/10.1007/978-3-319-96881-0_2
23. Gabizon, A., Williamson, Z.J., Ciobotaru, O.: PLONK: Permutations over lagrange-bases for oecumenical noninteractive arguments of knowledge. Cryptology ePrint Archive, Report 2019/953 (2019), https://eprint.iacr.org/2019/953
24. Ganesh, C., Khoshakhlagh, H., Parisella, R.: NIWI and New Notions of Extraction for Algebraic Languages. In: Galdi, C., Jarecki, S. (eds.) SCN 2022. LNCS, vol. 13409, pp. 687–710. Springer, Cham, Amalfi, Italy (September 12–14 2022)
25. Gennaro, R., Gentry, C., Parno, B., Raykova, M.: Quadratic span programs and succinct NIZKs without PCPs. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 626–645. Springer, Heidelberg (May 2013). https://doi.org/10.1007/978-3-642-38348-9_37
26. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: Fortnow, L., Vadhan, S.P. (eds.) 43rd ACM STOC. pp. 99–108. ACM Press (Jun 2011). https://doi.org/10.1145/1993636.1993651
27. González, A., Hevia, A., Ràfols, C.: QA-NIZK arguments in asymmetric groups: New tools and new constructions. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 605–629. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48797-6_25
28. González, A., Ràfols, C.: Shorter pairing-based arguments under standard assumptions. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 728–757. Springer, Heidelberg (Dec 2019). https://doi.org/10.1007/978-3-030-34618-8_25
29. González, A., Zacharakis, A.: Fully-succinct publicly verifiable delegation from constant-size assumptions. In: Nissim, K., Waters, B. (eds.) TCC 2021, Part I.

LNCS, vol. 13042, pp. 529–557. Springer, Heidelberg (Nov 2021). https://doi.org/10.1007/978-3-030-90459-3_18
30. Groth, J.: Short pairing-based non-interactive zero-knowledge arguments. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 321–340. Springer, Heidelberg (Dec 2010). https://doi.org/10.1007/978-3-642-17373-8_19
31. Groth, J.: On the size of pairing-based non-interactive arguments. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 305–326. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49896-5_11
32. Groth, J., Kohlweiss, M., Maller, M., Meiklejohn, S., Miers, I.: Updatable and universal common reference strings with applications to zk-SNARKs. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part III. LNCS, vol. 10993, pp. 698–728. Springer, Heidelberg (Aug 2018). https://doi.org/10.1007/978-3-319-96878-0_24
33. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432. Springer, Heidelberg (Apr 2008). https://doi.org/10.1007/978-3-540-78967-3_24
34. Hubacek, P., Wichs, D.: On the communication complexity of secure function evaluation with long output. In: Roughgarden, T. (ed.) ITCS 2015. pp. 163–172. ACM (Jan 2015). https://doi.org/10.1145/2688073.2688105
35. Ishai, Y., Weiss, M.: Probabilistically checkable proofs of proximity with zero-knowledge. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 121–145. Springer, Heidelberg (Feb 2014). https://doi.org/10.1007/978-3-642-54242-8_6
36. Izabachène, M., Libert, B., Vergnaud, D.: Block-wise P-signatures and non-interactive anonymous credentials with efficient attributes. In: Chen, L. (ed.) 13th IMA International Conference on Cryptography and Coding. LNCS, vol. 7089, pp. 431–450. Springer, Heidelberg (Dec 2011)
37. Jager, T., Rupp, A.: The semi-generic group model and applications to pairing-based cryptography. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 539–556. Springer, Heidelberg (Dec 2010). https://doi.org/10.1007/978-3-642-17373-8_31
38. Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from well-founded assumptions. In: Khuller, S., Williams, V.V. (eds.) 53rd ACM STOC. pp. 60–73. ACM Press (Jun 2021). https://doi.org/10.1145/3406325.3451093
39. Jain, A., Jin, Z.: Indistinguishability obfuscation via mathematical proofs of equivalence. In: 63rd FOCS. pp. 1023–1034. IEEE Computer Society Press (Oct / Nov 2022). https://doi.org/10.1109/FOCS54457.2022.00100
40. Jutla, C.S., Roy, A.: Shorter quasi-adaptive NIZK proofs for linear subspaces. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 1–20. Springer, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-42033-7_1
41. Kalai, Y.T., Paneth, O., Yang, L.: How to delegate computations publicly. In: Charikar, M., Cohen, E. (eds.) 51st ACM STOC. pp. 1115–1124. ACM Press (Jun 2019). https://doi.org/10.1145/3313276.3316411
42. Kalai, Y.T., Raz, R., Rothblum, R.D.: How to delegate computations: the power of no-signaling proofs. In: Shmoys, D.B. (ed.) 46th ACM STOC. pp. 485–494. ACM Press (May / Jun 2014). https://doi.org/10.1145/2591796.2591809
43. Kate, A., Zaverucha, G.M., Goldberg, I.: Constant-size commitments to polynomials and their applications. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 177–194. Springer, Heidelberg (Dec 2010). https://doi.org/10.1007/978-3-642-17373-8_11

44. Kilian, J.: A note on efficient zero-knowledge proofs and arguments (extended abstract). In: 24th ACM STOC. pp. 723–732. ACM Press (May 1992). https://doi.org/10.1145/129712.129782
45. Kiltz, E., Wee, H.: Quasi-adaptive NIZK for linear subspaces revisited. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 101–128. Springer, Heidelberg (Apr 2015). https://doi.org/10.1007/978-3-662-46803-6_4
46. Lai, R.W.F., Malavolta, G.: Subvector commitments with application to succinct arguments. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part I. LNCS, vol. 11692, pp. 530–560. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26948-7_19
47. Libert, B., Yung, M.: Concise mercurial vector commitments and independent zero-knowledge sets with short proofs. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 499–517. Springer, Heidelberg (Feb 2010). https://doi.org/10.1007/978-3-642-11799-2_30
48. Lindell, Y.: Parallel coin-tossing and constant-round secure two-party computation. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 171–189. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_10
49. Lipmaa, H.: Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 169–189. Springer, Heidelberg (Mar 2012). https://doi.org/10.1007/978-3-642-28914-9_10
50. Lipmaa, H.: Succinct non-interactive zero knowledge arguments from span programs and linear error-correcting codes. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 41–60. Springer, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-42033-7_3
51. Lipmaa, H.: A unified framework for non-universal SNARKs. In: Hanaoka, G., Shikata, J., Watanabe, Y. (eds.) PKC 2022, Part I. LNCS, vol. 13177, pp. 553–583. Springer, Heidelberg (Mar 2022). https://doi.org/10.1007/978-3-030-97121-2_20
52. Lipmaa, H., Pavlyk, K.: Gentry-Wichs is Tight: a Falsifiable Non-adaptively Sound SNARG. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021 (3). LNCS, vol. 13092, pp. 34–64. Springer, Cham, Singapore (Dec 5–9, 2021). https://doi.org/10.1007/978-3-030-92078-4_2
53. Lipmaa, H., Siim, J., Zajac, M.: Counting vampires: From univariate sumcheck to updatable ZK-SNARK. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022, Part II. LNCS, vol. 13792, pp. 249–278. Springer, Heidelberg (Dec 2022). https://doi.org/10.1007/978-3-031-22966-4_9
54. Okamoto, T., Pietrzak, K., Waters, B., Wichs, D.: New realizations of somewhere statistically binding hashing and positional accumulators. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 121–145. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48797-6_6
55. Paneth, O., Rothblum, G.N.: On zero-testable homomorphic encryption and publicly verifiable non-interactive arguments. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part II. LNCS, vol. 10678, pp. 283–315. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70503-3_9
56. Parno, B., Howell, J., Gentry, C., Raykova, M.: Pinocchio: Nearly practical verifiable computation. In: 2013 IEEE Symposium on Security and Privacy. pp. 238–252. IEEE Computer Society Press (May 2013). https://doi.org/10.1109/SP.2013.47

57. Råfols, C., Zapico, A.: An algebraic framework for universal and updatable SNARKs. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part I. LNCS, vol. 12825, pp. 774–804. Springer, Heidelberg, Virtual Event (Aug 2021). https://doi.org/10.1007/978-3-030-84242-0_27
58. Sahai, A., Waters, B.: How to use indistinguishability obfuscation: deniable encryption, and more. In: Shmoys, D.B. (ed.) 46th ACM STOC. pp. 475–484. ACM Press (May / Jun 2014). https://doi.org/10.1145/2591796.2591825
59. Waters, B., Wu, D.J.: Batch arguments for sfNP and more from standard bilinear group assumptions. In: Dodis, Y., Shrimpton, T. (eds.) CRYPTO 2022, Part II. LNCS, vol. 13508, pp. 433–463. Springer, Heidelberg (Aug 2022). https://doi.org/10.1007/978-3-031-15979-4_15
60. Zhandry, M.: To label, or not to label (in generic groups). In: Dodis, Y., Shrimpton, T. (eds.) CRYPTO 2022, Part III. LNCS, vol. 13509, pp. 66–96. Springer, Heidelberg (Aug 2022). https://doi.org/10.1007/978-3-031-15982-4_3
61. Zhang, C., Zhou, H.S., Katz, J.: An analysis of the algebraic group model. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022, Part IV. LNCS, vol. 13794, pp. 310–322. Springer, Heidelberg (Dec 2022). https://doi.org/10.1007/978-3-031-22972-5_11

A Additional Discussion

A.1 On Gentry-Wichs And QA-NIZKs

Some succinct QA-NIZKs achieve quasi-adaptive soundness for (specific) hard languages like linear subspaces [40,45] under falsifiable assumptions. Very recently, Campanelli et al. [12] clarify why this does not contradict Gentry-Wichs. As shown in [12], Gentry-Wichs only rules out reductions that cannot efficiently detect when the soundness property is broken. The language parameter of QA-NIZKs like [45] hides a trapdoor lt, such that knowing lt, one can efficiently decide membership of the statement in $\mathcal{L}_{\mathsf{lt}}$. That is, the language family is witness-sampleable. (See Section 3.2.) However, Punic is not witness-sampleable. Really, while Punic has a language trapdoor (the commitment extraction key), this is not sufficient to test if $\mathbb{x} \in \mathcal{L}_{\mathsf{lt}}$ or not — precisely since the statement (including the vector commitment) is succinct. Intuitively, in witness-sampleable languages, the statement $\mathbb{x}$ is sufficiently long so that the statement, the argument, and the trapdoor together encode the witness.

Similarly, in the case of SNARGs with an extractable commitment like DGPRS and FLPS [18,19], the reduction can black-box extract the witness from the commitment and use it to verify whether the input belongs to $\mathcal{L}_{\mathsf{lp}}$.

We will leave further study of this issue as an open problem.

A.2 Comparison with No-Signaling Approach

A recent approach to obtain global consistency out of local consistency uses no-signaling commitments [42,41,29,15]. In a nutshell, one divides the computation process into computation steps and then proves that (1) each step is correct (local consistency), (2) the consequent steps are consistent (global consistency).

For example, consider layered arithmetic circuits, with the layers being individual steps. (To connect it to what we are doing, note that each gate of the circuit corresponds to a constraint of R1CS.)

A natural approach is to commit to the state of the computation ($\mathsf{st}_1, \ldots, \mathsf{st}_T$) after each step and then prove (1) and (2) on the committed values. To prove both (1) and (2), [29,15] propose to use SE commitment schemes. However, their use of SE commitment schemes is different from ours. While we allow black-box somewhere extraction of the wire values neighboring a single gate, the basic approach of [29,15] uses the commitment scheme to black-box extract values pertaining to two computation states. The no-signaling property of commitments is used to prove consistency between different black-box extractions. Intuitively, no-signaling guarantees that if one extracts $\mathcal{S}_0 = (\mathsf{st}_{i-1}, \mathsf{st}_i)$ with one CRS and then $\mathcal{S}_1 = (\mathsf{st}_i, \mathsf{st}_{i+1})$ with another CRS, then the value of $\mathcal{S}_0 \cap \mathcal{S}_1 = \mathsf{st}_i$ is unchanged. (See [29] for discussion.) In the case of arithmetic circuits, this means extracting all wire values in two neighboring gate layers. The latter is only practical if the circuit has *bounded width*. ([15] considers Turing machines instead of circuits, and in this case, one needs to have *bounded space*.)

While methods are known to move from bounded width or bounded space to general computation, they make the solution more costly. For example, [41,15] considers the following approach. Simulate a Turing machine $\mathcal{M}$ with large space via a RAM machine $\mathfrak{R}$, where the RAM machine has access to large untrusted external memory but small internal memory. A digest of the external memory, in the form of the root of the hash tree, is stored in the internal memory. The prover shows that $\mathfrak{R}$ starting at configuration s (including the large external memory) transitions to configuration t in T steps where the verifier is only given digests h_s and h_t of the two configurations. See [41,15].

Compared to that, we can directly handle general computation. Since we use SE commitments to black-box extract only a neighborhood of a single gate, we are restricted to limiting the fan-in and fan-out of all gates but not the width of the circuit. Well-known standard transformations can be used to limit the fan-in and fan-out with essentially minimal overhead. Due to that, the new zk-SNARK is concretely efficient in every aspect. On the other hand, no such transformations are known that limit the circuit width.

Our approach to achieving global consistency has direct advantages compared to the approach of no-signaling commitments. We leave it as an open question whether one can construct SNARGs for P using our approach.

A.3 On Kilian's Zero-Knowledge Argument

One can construct a relatively simple but inefficient semi-adaptive black-box knowledge-sound zk-SNARG based on Kilian's seminal interactive zero-knowledge argument [44]. Recall that in the latter, the verifier first chooses a hash function key hk, the prover then (hash tree-)commits to the PCP-transformed witness, the verifier makes some queries to the PCP based on fresh randomness ϱ, and the prover answers to the queries. (The prover's answer also includes

corresponding hash certificates.) In the resulting black-box knowledge-sound zk-SNARG, hk would be the language parameter, the prover's commitment would be the statement (that includes the public input), the verifier's randomness ϱ would be the CRS, and the query answers would be the SNARG argument. For the soundness reduction to go through, the PCP must be knowledge-sound, and the hash function must be collision-resistant; see [46] for a discussion and a reduction in the interactive case. To get zero knowledge, the PCP must be a ZK-PCP [35].

Kilian-based solution has direct correspondence to Punic: for example, the hash tree plays the role of vector commitments, the verifier randomness ϱ defines a constraint with its neighborhood, the witness can be extracted after sufficiently many iterations (by using the knowledge-soundness property of the PCP), and the hash certificates can be used to obtain a collision when there is no full witness. For greater efficiency (though that would result in worse assumptions), the hash tree can be replaced with a subvector commitment [46]. This means that the Kilian-based solution can be based on a wide array of computational assumptions and trust models.

Unfortunately, PCPs are *very* inefficient [7] (e.g., the PCP proof length is at least $\Theta(n \log^3 n)$), and thus such a solution only has a theoretical value. Because of this, modern zk-SNARKs are based on interactive oracle proofs (IOPs, [8]) that are multi-round alternatives to PCPs. Since (falsifiable) semi-adaptive SNARGs have round constraints, IOPs cannot be used in our application. To our knowledge, Punic is the only round-efficient alternative to PCP-based SNARGs.

B Additional Preliminaries

B.1 The QA-SNARG argument system BLS

For the sake of completeness, we depict the González-Hevia-Ràfols bilateral subspace QA-SNARG argument system BLS for $\mathcal{L}_{\mathsf{lp}}$ in Fig. 8 (see Appendix B.1).

B.2 QALINRES: Additional Background

Motivation behind QALINRES. DGPRS and FLPS are based on TSDH-like [56] assumptions: n-STDSH and n-QTSDH in [18] and n-SATSDH in [19]. These assumptions were tailored to the SSP and SAP constraint systems, used in [18] and [19], correspondingly. To be able to use R1CS, [52] first defined a new assumption (n-QATSDH) and then argued why QALINRES is simpler.

As motivated by Lipmaa and Pavlyk [52], all TSDH-type assumptions have one serious problem. Namely, to argue that such assumptions are sensible, one can prove that they hold in the *generic group model* (GGM). In a GGM proof, one considers a generic adversary that is only allowed to (i) execute group operations in the source and target groups, (ii) perform the pairing operation, and (iii) check for equality of two group elements. GGM is a very restrictive model.

$\mathsf{BLS.K_{lp}}(\mathsf{p})$: $\left([\boldsymbol{M}]_*, \binom{\boldsymbol{M}_1}{\boldsymbol{M}_2}\right) \leftarrow\!\$\ \mathsf{K_{lt}}(\mathsf{p})$; **return** $\mathsf{lp} \leftarrow [\boldsymbol{M}]_*$;
$\mathsf{BLS.K_{crs}}(\mathsf{lp} = [\boldsymbol{M}]_*)$: $\bar{\boldsymbol{A}} \leftarrow\!\$\ \mathcal{D}_\kappa$; // $\boldsymbol{A} \in \mathbb{F}^{(\kappa+1)\times\kappa}$, $\bar{\boldsymbol{A}}$ is invertible
$\boldsymbol{K}_1 \leftarrow\!\$\ \mathbb{F}^{n_1\times\kappa}$; $\boldsymbol{K}_2 \leftarrow\!\$\ \mathbb{F}^{n_2\times\kappa}$; $\boldsymbol{\Delta} \leftarrow\!\$\ \mathbb{F}^{\kappa\times m}$;
$\boldsymbol{C}_1 \leftarrow \boldsymbol{K}_1\bar{\boldsymbol{A}}$; $\boldsymbol{C}_2 \leftarrow \boldsymbol{K}_2\bar{\boldsymbol{A}}$; // $\boldsymbol{C}_\gamma \in \mathbb{F}^{n_\gamma\times\kappa}$
$[\boldsymbol{P}_1]_1 \leftarrow \boldsymbol{K}_1^\intercal[\boldsymbol{M}_1]_1 + [\boldsymbol{\Delta}]_1$; $[\boldsymbol{P}_2]_2 \leftarrow \boldsymbol{K}_2^\intercal[\boldsymbol{M}_2]_2 - [\boldsymbol{\Delta}]_2$; // $[\boldsymbol{P}_\gamma]_\gamma \in \mathbb{G}_\gamma^{\kappa\times m}$
$\mathsf{crs} \leftarrow ([\bar{\boldsymbol{A}}, \boldsymbol{C}_2, \boldsymbol{P}_1]_1, [\bar{\boldsymbol{A}}, \boldsymbol{C}_1, \boldsymbol{P}_2]_2)$; $\mathsf{td} \leftarrow (\boldsymbol{K}_1, \boldsymbol{K}_2)$; **return** $(\mathsf{crs}, \mathsf{td})$;
$\mathsf{BLS.P}(\mathsf{lp}, \mathsf{crs}; ([\mathbf{c}_1]_1, [\mathbf{c}_2]_2), \mathbb{w})$: $\boldsymbol{r}_\pi \leftarrow\!\$\ \mathbb{F}^\kappa$;
$[\boldsymbol{\pi}_1]_1 \leftarrow [\boldsymbol{P}_1]_1\mathbb{w} + [\boldsymbol{r}_\pi]_1$; $[\boldsymbol{\pi}_2]_2 \leftarrow [\boldsymbol{P}_2]_2\mathbb{w} - \boldsymbol{r}_\pi[1]_2$; // $[\boldsymbol{\pi}_\gamma]_\gamma \in \mathbb{G}_\gamma^\kappa$
return $\pi \leftarrow ([\boldsymbol{\pi}_1]_1, [\boldsymbol{\pi}_2]_2)$;
$\mathsf{BLS.V}(\mathsf{lp}, \mathsf{crs}; ([\mathbf{c}_1]_1, [\mathbf{c}_2]_2), \pi)$: // Equality check is done in $\mathbb{G}_T^{1\times\kappa}$
return $[\mathbf{c}_1]_1^\intercal \bullet [\boldsymbol{C}_1]_2 + [\mathbf{c}_2]_2^\intercal \bullet [\boldsymbol{C}_2]_1 \stackrel{?}{=} [\boldsymbol{\pi}_1]_1^\intercal \bullet [\bar{\boldsymbol{A}}]_2 + [\boldsymbol{\pi}_2]_2^\intercal \bullet [\bar{\boldsymbol{A}}]_1$;
$\mathsf{BLS.Sim}(\mathsf{lp}, \mathsf{crs}, \mathsf{td}, ([\mathbf{c}_1]_1, [\mathbf{c}_2]_2))$: $\boldsymbol{r}'_\pi \leftarrow\!\$\ \mathbb{F}^\kappa$; // $[\mathbf{c}_\gamma]_\gamma \in \mathbb{G}_\gamma^{n_\gamma}$
$[\boldsymbol{\pi}'_1]_1 \leftarrow \boldsymbol{K}_1^\intercal[\mathbf{c}_1]_1 + [\boldsymbol{r}'_\pi]_1$; $[\boldsymbol{\pi}'_2]_2 \leftarrow \boldsymbol{K}_2^\intercal[\mathbf{c}_2]_2 - [\boldsymbol{r}'_\pi]_2$; // $[\boldsymbol{\pi}'_\gamma]_\gamma \in \mathbb{G}_\gamma^\kappa$
return $\pi' \leftarrow ([\boldsymbol{\pi}'_1]_1, [\boldsymbol{\pi}'_2]_2)$;

Fig. 8. The bilateral subspace QA-SNARG BLS.

One of the many criticisms against GGM is that the target group $\mathbb{G}_T$ is a subgroup of the finite field, and thus it is questionable whether it can be modeled as a generic group, [37]. Indeed, one can use the finite field structure to operate on the elements of the $\mathbb{G}_T$. To address this issue, [37] defined the *semi-GGM*, where one assumes that only the source groups are generic. A significant drawback of STDSH, QTSDH, SATDSH, and QATSDH is that, in their definition, the adversary can output a value in the target group. Thus, they are not (known to be) secure in the semi-GGM.

Because of this, [52] defined the QALINRES assumption where the adversary does not output $\mathbb{G}_T$ elements. Moreover, [52] also shows that QALINRES is a particular case of the QATSDH assumption. An additional benefit of QALINRES is that it does not force the adversary to output pairs of elements like $(\eta_\mathsf{a}, \hat{\eta}_\mathsf{a} = \eta_\mathsf{a} y)$; thus, arguments that use QALINRES are potentially more efficient than arguments that use QATSDH. See [52] for more information.

B.3 On Punic and SSP

SSP. One can simplify Punic significantly by targeting SSP [17] (i.e., Boolean circuits) instead of R1CS [25] (i.e., arithmetic circuits). If the SSP constraints are satisfied, all wire values are Boolean. Thus, in the special soundness proof, the black-box extractor $\mathsf{Ext_{ss}}$ extracts $G(\mathbb{w}) = [\mathbb{w}y]_1$ for Boolean $\mathbb{w}_j$. Since $\mathbb{w}_j$ is Boolean, one can efficiently recover $\mathbb{w}$, achieving black-box *id*-knowledge-soundness for the identity map *id*. Transferring Punic to the SSP setting has other benefits. For example, we could use the CDHK vector commitment scheme and a standard linear subspace argument [45] instead of VCF and the bilateral linear subspace argument. The SNARK itself would simplify, and QALINRES (see Definition 1) would become a more standard-looking assumption.

Since this comes at the cost of handling SSP (Boolean circuits), we chose to present a SNARK for R1CS. However, any SNARK for R1CS can be easily modified to a SNARK for SSP due to an observation of [51] that an SSP instance is an R1CS instance but with all three R1CS matrices $\boldsymbol{U}$, $\boldsymbol{V}$, $\boldsymbol{W}$ being equal. In addition, an SSP constraint system must have a new constraint for every wire, enforcing the wire value to be Boolean.

C AGM + PDL ⇒ QALINRES

C.1 Preliminaries

Let $n_1(\lambda), n_2(\lambda) \in \mathsf{poly}(\lambda)$. Pgen is $(n_1(\lambda), n_2(\lambda))$-PDL *(Power Discrete Logarithm) secure* if for any λ and non-uniform PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{pdl}}_{n_1,n_2,\mathsf{Pgen},\mathcal{A}}(\lambda) :=$

$$\Pr\left[\mathcal{A}(\mathsf{p}, [(x^i)_{i=0}^{n_1}]_1, [(x^i)_{i=0}^{n_2}]_2) = x \mid \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda), x \leftarrow\!\$\, \mathbb{F}\right] \approx_\lambda 0 \ .$$

Algebraic Group Model. AGM [22] is a recent idealized model of computation. Essentially, in the AGM, one assumes that each PPT algorithm $\mathcal{A}$ is algebraic in the following sense. Assume $\mathcal{A}$'s input includes $[\mathbb{x}_\gamma]_\gamma$ and no other elements from the group $\mathbb{G}_\gamma$. We assume that if $\mathcal{A}$ outputs a vector $[\mathbb{s}_\gamma]_\gamma$ of group elements, then $\mathcal{A}$ knows a matrix $\boldsymbol{\gamma}_\gamma$, such that $\mathbb{s}_\gamma = \boldsymbol{\gamma}_\gamma^\intercal \mathbb{x}_\gamma$.

Fix Pgen. More precisely, a PPT algorithm $\mathcal{A}$ is *algebraic* if there exists an efficient extractor $\mathsf{Ext}_\mathcal{A}$, such that for any vector of group elements $\mathbb{x} = ([\mathbb{x}_1]_1, [\mathbb{x}_2]_2)$, $\mathsf{Adv}^{\mathrm{agm}}_{\mathsf{Pgen},\mathcal{A},\mathsf{Ext}_\mathcal{A}}(\lambda) :=$

$$\Pr\left[\begin{array}{l|c} \mathbb{s}_1 \neq \boldsymbol{\gamma}_1^\intercal \mathbb{x}_1 \vee & \mathsf{p} \leftarrow\!\$\, \mathsf{Pgen}(1^\lambda); r \leftarrow\!\$\, \mathsf{RND}_\lambda(\mathcal{A}); \\ \mathbb{s}_2 \neq \boldsymbol{\gamma}_2^\intercal \mathbb{x}_2 & ([\mathbb{s}_1]_1, [\mathbb{s}_2]_2) \leftarrow\!\$\, \mathcal{A}(\mathsf{p}, \mathbb{x}; r); (\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2) \leftarrow \mathsf{Ext}_\mathcal{A}(\mathbb{x}; r) \end{array}\right] \approx_\lambda 0 \ .$$

C.2 Security Theorem

Theorem 5. *AGM +* (n, n)-PDL ⇒ n-QALINRES.

Proof. Let $\boldsymbol{X} = (X, Y)$ be two indeterminates and let (x, y) be the corresponding trapdoors. Denote

$$\mathsf{ck} = (([x^i]_1, [x^i]_2)_{i=0}^n, [y]_1, [y]_2) \ .$$

Let $\mathcal{A}$ be an algebraic n-QALINRES adversary that has a success probability $\varepsilon_\mathcal{A}$ in breaking QALINRES. Assume that we are now in the case when $\mathcal{A}$ succeeds. That is, given $(\mathsf{ck}; r)$ as an input, $\mathcal{A}$ outputs a tuple

$$\pi = (\mathsf{j}, [\mathsf{a}, \hat{\eta}_\mathsf{a}, \varphi_\mathsf{a}, \mathsf{c}, \hat{\eta}_\mathsf{c}, \varphi_\mathsf{c}, \mathsf{h}]_1, [\mathsf{b}, \hat{\eta}_\mathsf{b}, \varphi_\mathsf{b}]_2) \ ,$$

such that all five conditions in Definition 1 hold.

Since $\mathcal{A}$ is algebraic, there exists an extractor $\mathsf{Ext}_\mathcal{A}$ that, on the same inputs, succeeds with the probability

$$1 - \mathsf{Adv}^{\mathrm{agm}}_{\mathsf{Pgen},\mathcal{D},\mathcal{A},\mathsf{Ext}_\mathcal{A}}(\lambda) = 1 - \mathsf{negl}(\lambda) \ .$$

If $\mathsf{Ext}_{\mathcal{A}}$ succeeds, then taking into account that elements of ck consist of known polynomials in (X, Y) and one can efficiently compute the coefficients of polynomials like $\mathsf{a}(X, Y)$ from the output of $\mathsf{Ext}_{\mathcal{A}}$, we write

$$T(X, Y) = T_x(X) + T_y Y \ ,$$

where either

$$T \in S := \{\mathsf{a}, \hat{\eta}_{\mathsf{a}}, \varphi_{\mathsf{a}}, \mathsf{c}, \hat{\eta}_{\mathsf{c}}, \varphi_{\mathsf{c}}, \mathsf{h}, \mathsf{b}, \hat{\eta}_{\mathsf{b}}, \varphi_{\mathsf{b}}\} \ .$$

Here, the polynomials $T_x(X)$ have degree $\leq n$ while T_y are field elements. For example, $\mathsf{a}(X, Y) = \mathsf{a}_x(X) + \mathsf{a}_y Y$ for $\deg \mathsf{a}_x(X) \leq n$. Thus, $\mathsf{a}(X, Y)$, $\varphi_{\mathsf{a}}(X, Y)$, $\ldots$, $\varphi_{\mathsf{b}}(X, Y)$ are maliciously chosen polynomials, such that say $\mathsf{a} = \mathsf{a}(X, Y) = \mathsf{a}_x(X) + \mathsf{a}_y Y$.

Following the strategy of [22], we construct a PDL adversary $\mathcal{B}$. (See Fig. 9 for the description of $\mathcal{B}$.) $\mathcal{B}$ has inputs depending on a single trapdoor x. $\mathcal{B}$ implicitly creates another random trapdoor $y \leftarrow sx + t$, and uses it to create a valid input for $\mathcal{A}$. $\mathcal{B}$ then runs $\mathcal{A}$. After that, $\mathcal{B}$ uses $\mathsf{Ext}_{\mathcal{A}}$ to extract the coefficients of various polynomials $T(X, Y)$. Thus, for $\mathcal{B}$, all values returned by $\mathcal{A}$ are univariate polynomials (in indeterminate X, corresponding to trapdoor x). Based on extracted polynomials $T(X, Y)$, $\mathcal{B}$ obtains all coefficients of five verification polynomials $V_i(X, Y)$, where

$$\begin{aligned}
V_1(X, Y) &= \mathsf{a}(X, Y)Y - (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}}(X, Y)Y + \hat{\eta}_{\mathsf{a}}(X, Y) \ , \\
V_2(X, Y) &= \mathsf{b}(X, Y)Y - (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{b}}(X, Y)Y + \hat{\eta}_{\mathsf{b}}(X, Y) \ , \\
V_3(X, Y) &= \mathsf{c}(X, Y)Y - (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{c}}(X, Y)Y + \hat{\eta}_{\mathsf{c}}(X, Y) \ , \\
V_4(X, Y) &= \mathsf{a}(X, Y)\mathsf{b}(X, Y) - \mathsf{c}(X, Y) - \mathsf{h}(X, Y)Z_{\mathbb{H}}(X) \ , \\
V_5(X, Y) &= \hat{\eta}_{\mathsf{a}}(X, Y)\hat{\eta}_{\mathsf{b}}(X, Y) - \hat{\eta}_{\mathsf{c}}(X, Y)Y.
\end{aligned} \tag{9}$$

By inspecting the winning conditions of a QALINRES adversary (see Definition 1), the QALINRES verifier checks that $[V_1(x, y)]_1 = [0]_1$, $[V_2(x, y)]_2 = [0]_2$, $[V_3(x, y)]_1 = [0]_1$, $[V_4(x, y)]_T = [0]_T$, and $[V_5(x, y)]_T \neq [0]_T$.

Since $\mathcal{B}$ created y implicitly as affine polynomials of x, from his viewpoint each $V_i(X, Y)$ is a known univariate polynomial $V_i'(X) = V_i(X, sX + t)$. We will analyze later the case $\mathcal{B}$ aborts at step (*) (see Fig. 9), i.e., $V_i(X, Y) = 0$ for each $i \leq 4$, and show it can happen only with negligible probability.

Assume that $\mathcal{B}$ did not abort. Then, for some $i \in [1, 4]$, $V_i(X, Y) \neq 0$ and $V_i(x, y) = 0$. Since y is a function of x, we get that $V_i'(X) \neq 0$ but $V_i'(x) = 0$. $\mathcal{B}$ uses a polynomial factorization algorithm to find all roots of $V_i'(X)$, and one of them has to be equal to y. Thus, $\mathcal{B}$ has broken the PDL assumption with probability $\varepsilon_{\mathcal{A}} - \mathsf{negl}(\lambda)$. (See Fig. 9.)

Analysis of abortion probability in (*). Assume now that $V_i(X, Y) = 0$, for $i \leq 4$, and $V_5(X, Y) \neq 0$ (all as polynomials). Since $V_1(X, Y) = 0$,

$$\begin{aligned}
\mathsf{a}(X, Y) &= \mathsf{a}_x(X) + \mathsf{a}_y Y \\
&= \left(\varphi_{\mathsf{a}x}(X) + (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}y} Y\right) + \left(\hat{\eta}_{\mathsf{a}x}(X) + \hat{\eta}_{\mathsf{a}y} Y\right)/Y \ .
\end{aligned}$$

$\mathcal{B}(\mathsf{p}, \mathbb{x}_{\mathcal{B}} = ([x^i]_1, [x^i]_2)_{i=0}^n)$

$s, t \leftarrow\!\!\$\ \mathbb{Z}_p^*$; $[y]_1 \leftarrow s[x]_1 + t[1]_1$; $[y]_2 \leftarrow s[x]_2 + t[1]_2$;
$\mathsf{ck} \leftarrow ([(x^i)_{i=0}^n, y]_1, [(x^i)_{i=0}^n, y]_2)$;
$r \leftarrow\!\!\$\ \mathsf{RND}_\lambda(\mathcal{A})$; $\pi \leftarrow \mathcal{A}(\mathsf{ck}; r)$;
$(T_x(X), T_y)_{T \in S} \leftarrow \mathsf{Ext}_{\mathcal{A}}(\mathsf{ck}; r)$;
If $\mathsf{Ext}_{\mathcal{A}}$ fails, then abort;
If $V_1(X,Y) = \ldots = V_4(X,Y) = 0$, then abort (*);
Let $i \le 4$ be such that $V_i(X,Y) \neq 0$;
Obtain roots x_k of $V_i'(X)$;
Return x_k that satisfies $[x_k]_1 = [x]_1$;

Fig. 9. QALINRES $\Rightarrow$ PDL reduction $\mathcal{B}$.

Let $R = \mathbb{F}[X]$. Think of $V_1(X,Y)$ as a polynomial over $R[Y]$, $V_1(X,Y) = \sum_i V_{1i}(X)Y^i \in R[Y]$. Since $V_1(X,Y) = 0$, each of its R-coefficients has to be zero. Consider next the implications:

- $V_{11} = 0$ (the coefficient of Y in $V_1(X,Y)$ is 0): thus,
$$\mathsf{a}_x(X) = (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}x}(X) + \hat{\eta}_{\mathsf{a}y} \ .$$
- $V_{12} = 0$ (the coefficient of Y^2 in $V_1(X,Y)$ is 0): thus, $\mathsf{a}_y = (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}y}$. Since a_y and $\varphi_{\mathsf{a}y}$ are both field elements, $\varphi_{\mathsf{a}y} = \mathsf{a}_y = 0$.
- $V_{10} = 0$ (the coefficient of 1 in $V_1(X,Y)$ is 0): thus, $\hat{\eta}_{\mathsf{a}x}(X) = 0$.
Thus, $\varphi_{\mathsf{a}}(X,Y) = \varphi_{\mathsf{a}x}(X)$, $\hat{\eta}_{\mathsf{a}}(X,Y) = \hat{\eta}_{\mathsf{a}y}Y$, and thus, by $V_1(X,Y) = 0$,
$$\mathsf{a}(X,Y) = (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}x}(X) + \hat{\eta}_{\mathsf{a}y} \ .$$

Analogously, from $V_2(X,Y) = 0$ and $V_3(X,Y) = 0$, we get

$$\mathsf{b}(X,Y) = (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{b}x}(X) + \hat{\eta}_{\mathsf{b}y} \ ,$$
$$\mathsf{c}(X,Y) = (X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{c}x}(X) + \hat{\eta}_{\mathsf{c}y} \ .$$

But then from $V_4(X,Y) = 0$, we then get that

$$V_4(X,Y) := \left((X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}x}(X) + \hat{\eta}_{\mathsf{a}y}\right)\left((X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{b}x}(X) + \hat{\eta}_{\mathsf{b}y}\right) - \left((X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{c}x}(X) + \hat{\eta}_{\mathsf{c}y}\right) - (\mathsf{h}_x(X) + h_y Y)\, Z_{\mathbb{H}}(X) = 0 \ .$$

Since the coefficient of Y in $V_4(X,Y) \in R[Y]$ is $h_y Z_{\mathbb{H}}(X)$ and $Z_{\mathbb{H}}(X) \neq 0$, we finally get $h_y = 0$. Thus,

$$V_4(X,Y) = ((X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{a}x}(X) + \hat{\eta}_{\mathsf{a}y})((X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{b}x}(X) + \hat{\eta}_{\mathsf{b}y}) - ((X - \omega^{\mathsf{j}-1})\varphi_{\mathsf{c}x}(X) + \hat{\eta}_{\mathsf{c}y}) - \mathsf{h}_x(X)Z(X) = 0 \ .$$

Since $V_4(X,Y) = 0$, then also $V_4(X,Y) \equiv 0 \pmod{X - \omega^{\mathsf{j}-1}}$. Hence, $\hat{\eta}_{\mathsf{a}y}\hat{\eta}_{\mathsf{b}y} = \hat{\eta}_{\mathsf{c}y}$, a contradiction with

$$V_5'(Y) = \hat{\eta}_{\mathsf{a}}(x,Y)\hat{\eta}_{\mathsf{b}}(x,Y) - \hat{\eta}_{\mathsf{c}}(x,Y)Y = \hat{\eta}_{\mathsf{a}y}\hat{\eta}_{\mathsf{b}y} - \hat{\eta}_{\mathsf{c}y} \neq 0 \ .$$

Hence, if $\mathcal{A}$ is honest, then $\mathcal{B}$ never aborts. □

$\mathcal{B}_{\mathsf{qal}}(\mathsf{ck})$ // $\mathsf{ck} = ([(x^i)_{i=0}^n, y]_1, [(x^i)_{i=0}^n, y]_2)$

$(j, [\hat{\eta}, \varphi]_1) \leftarrow \mathcal{A}(\mathsf{ck}); [\mathsf{a}]_1 \leftarrow [1]_1; [\mathsf{b}]_2 \leftarrow [1]_2; [\mathsf{c}]_1 \leftarrow [1]_1;$
return $(j, [\mathsf{a}, y - \hat{\eta}, \varphi, \mathsf{c}, y, 0, 0]_1, [\mathsf{b}, y, 0]_2);$

Fig. 10. QALINRES adversary in Lemma 3.

D Missing Lemmas And Proofs

D.1 Proof of Lemma 3

Proof. Let $\mathcal{A}$ be a VCSDH adversary (Definition 4). In Fig. 10, we depict an adversary $\mathcal{B}_{\mathsf{qal}}$ that uses $\mathcal{A}$ to break QALINRES (Definition 1). When $\mathcal{A}$ succeeds, $0 = (x - \omega^{j-1})\varphi - \hat{\eta}/y$ for $\hat{\eta} \neq 0$. $\mathcal{B}_{\mathsf{qal}}$ sets $\mathsf{a} = \mathsf{b} = \mathsf{c} = 1$. Thus,

$$\mathsf{a} = 1 = (x - \omega^{j-1})\varphi + (y - \hat{\eta})/y$$

for $\hat{\eta} \neq 0$. Clearly, $\mathsf{b} = (x - \omega^{j-1}) \cdot 0 + 1$, $\mathsf{c} = (x - \omega^{j-1}) \cdot 0 + 1$, $\mathsf{ab} - \mathsf{c} = 0 = \mathsf{h} Z_{\mathbb{H}}(x)$ (since $\mathsf{h} = 0$), and $(y - \hat{\eta}) \cdot y \neq y \cdot y$. Thus, $\mathcal{B}_{\mathsf{qal}}$ succeeds. □

D.2 Proof of Lemma 4

Proof. Clearly, $[\boldsymbol{U}_j \boldsymbol{\mu}]_1 = [\mathsf{ck}_{\boldsymbol{e},j}]_1 \left(\begin{smallmatrix} \boldsymbol{\mu} \\ r \end{smallmatrix}\right)$ and

$$[C(x)]_1 = [\mathsf{ck}_\ell]_1 \left(\begin{smallmatrix} \boldsymbol{U}\boldsymbol{\mu} \\ r \end{smallmatrix}\right) = [\mathsf{ck}_\ell]_1 \left(\begin{smallmatrix} \boldsymbol{U} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{smallmatrix}\right) \left(\begin{smallmatrix} \boldsymbol{\mu} \\ r \end{smallmatrix}\right) = [\mathsf{ck}_u]_1 \left(\begin{smallmatrix} \boldsymbol{\mu} \\ r \end{smallmatrix}\right) \ .$$

Moreover,

$$\begin{aligned} Q_{u_k,j}(X) &= \frac{u_k(X) - u_k(\omega^{j-1})}{X - \omega^{j-1}} = \frac{\sum_{i=1}^n U_{ik}(\ell_i(X) - \ell_i(\omega^{j-1}))}{X - \omega^{j-1}} \\ &= \sum_{i=1}^n U_{ik} \frac{\ell_i(X) - \ell_i(\omega^{j-1})}{X - \omega^{j-1}} = \sum_{i=1}^n U_{ik} Q_{\ell_i,j}(X) \ . \end{aligned}$$

Thus,

$$\mathsf{ck}_{\ell,j} \left(\begin{smallmatrix} \boldsymbol{U}\boldsymbol{\mu} \\ r \end{smallmatrix}\right) = \sum_{i=1}^n (\boldsymbol{U}\boldsymbol{\mu})_i Q_{\ell_i,j}(x) = \sum_{k=1}^m \mu_k Q_{u_k,j}(x) + r Q_{Z_{\mathbb{H}},j}(x) = \mathsf{ck}_{u,j} \left(\begin{smallmatrix} \boldsymbol{\mu} \\ r \end{smallmatrix}\right) \ .$$

□

D.3 Proof of Theorem 1

Proof. Let $\mathcal{A}_{\mathsf{ks}}$ be a knowledge-soundness adversary that, in the first game (Game_1) of Definition 5, succeeds in convincing $\mathcal{D}$ with a non-negligible probability $\varepsilon_1(\lambda)$. If $\varepsilon_1(\lambda)$ is negligible, there is nothing to prove: since $\varepsilon_2(\lambda) \leq \varepsilon_1(\lambda)$,

$\mathcal{A}_{\mathsf{ss}}^{\mathcal{A}_2(\mathsf{st},\cdot)}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}})$

1 : **for** $\varrho \in [1,n]$ **do**
2 : $\quad\mathcal{T} \leftarrow \emptyset$;
3 : $\quad$**while** $\mathcal{T} \neq \mathbb{F}$ **do**
4 : $\quad\quad r \leftarrow\!\!\$\ \mathbb{F} \setminus \mathcal{T}; \mathcal{T} \leftarrow \mathcal{T} \cup \{r\}$;
5 : $\quad\quad(\mathsf{crs}^{\varrho}, \mathsf{td}^{\varrho}) \leftarrow \mathsf{K}_{\mathsf{crs}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \varrho; r)$;
6 : $\quad\quad\pi^{\varrho} \leftarrow \mathcal{A}_2(\mathsf{st}, \mathsf{crs}^{\varrho})$;
7 : $\quad\quad$**if** $\mathsf{V}(\mathsf{lp}, \mathsf{crs}^{\varrho}, C, \mathbb{x}, \pi^{\varrho}) = 1$
8 : $\quad\quad$ **then break**; **fi**
9 : $\quad$**endwhile**
10 : $\quad$**if** $\mathsf{V}(\mathsf{lp}, \mathsf{crs}^{\varrho}, C, \mathbb{x}, \pi^{\varrho}) = 0$
11 : $\quad$ **then return** $\bot$; **fi**
12 : $\quad\mathsf{tr}^{\varrho} \leftarrow (C, \mathbb{x}, \mathsf{crs}^{\varrho}, \mathsf{td}^{\varrho}, \pi^{\varrho})$;
13 : **endfor**
14 : **return** $\mathbf{tr} \leftarrow (\mathsf{tr}^{\varrho})_{\varrho=1}^{n}$;

$\mathsf{Ext}_{\mathsf{ks}}^{\mathcal{A}_2(\mathsf{st},\cdot)}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},C,\mathbb{x},\mathsf{st})$

1 : $(\mathsf{crs},\mathsf{td}) \leftarrow\!\!\$\ \mathsf{K}_{\mathsf{crs}}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}},0)$;
2 : $\pi \leftarrow \mathcal{A}_2(\mathsf{st},\mathsf{crs})$;
3 : **if** $\mathsf{V}(\mathsf{lp},\mathsf{crs},C,\mathbb{x},\pi) = 0$
4 : $\quad$ **then return** $\bot$; **fi**
5 : $\mathsf{tr}_0 \leftarrow (C,\mathbb{x},\mathsf{crs},\mathsf{td},\pi)$;
6 : $\mathbf{tr} \leftarrow \mathcal{A}_{\mathsf{ss}}^{\mathcal{A}_2(\mathsf{st},\cdot)}(\mathsf{lp},\mathcal{R}_{\mathfrak{J}})$;
7 : **if** $\mathbf{tr} = \bot$ **then return** $(\mathsf{tr}_0,\bot)$; **fi**
8 : $G(\mathbb{w}) \leftarrow \mathsf{Ext}_{\mathsf{ss}}(\mathsf{lp},\mathbf{tr})$;
9 : **return** $(\mathsf{tr}_0, G(\mathbb{w}))$;

Fig. 11. $\mathcal{A}_{\mathsf{ss}}$ and $\mathsf{Ext}_{\mathsf{ks}}$ in the second game of Theorem 1.

$|\varepsilon_2(\lambda) - \varepsilon_1(\lambda)| \approx_\lambda 0$. Assume Π is CRS-indistinguishable. Thus, for any PPT $\mathcal{A}_{\mathsf{ind}}$, $\varepsilon'(\lambda) := \mathsf{Adv}^{\mathrm{crsind}}_{\mathsf{Pgen},\varrho,\Pi,\mathcal{A}_{\mathsf{ind}}}(\lambda)$ is negligible.

Consider the second game in Definition 5 that defines $\varepsilon_2(\lambda)$. Let $\mathsf{Ext}_{\mathsf{ss}}$ be a PPT special soundness extractor. In Fig. 11, we depict a special soundness adversary $\mathcal{A}_{\mathsf{ss}}$ and a black-box knowledge-soundness extractor $\mathsf{Ext}_{\mathsf{ks}}$. $\mathcal{A}_{\mathsf{ss}}$ loops over $\varrho \in [1,n]$. For each ϱ, we have an inner loop, where $\mathcal{A}_{\mathsf{ss}}$ samples (without replacement) a randomizer r to construct crs^{ϱ}. The inner loop ends when $\mathcal{A}_2$ produces an acceptable argument π^{ϱ}. If no acceptable argument was found for some ϱ, $\mathcal{A}_{\mathsf{ss}}$ aborts. Otherwise, $\mathcal{A}_{\mathsf{ss}}$ outputs an n-tuple of acceptable arguments.

$\mathsf{Ext}_{\mathsf{ks}}$ runs $\mathsf{K}_{\mathsf{crs}}$ and the adversary once with $\varrho = 0$ and obtains a transcript tr_0. $\mathsf{Ext}_{\mathsf{ks}}$ aborts when tr_0 is not acceptable. Otherwise, $\mathsf{Ext}_{\mathsf{ks}}$ calls $\mathcal{A}_{\mathsf{ss}}$ to obtain n more transcripts. $\mathsf{Ext}_{\mathsf{ks}}$ aborts when $\mathcal{A}_{\mathsf{ss}}$ aborted. Otherwise, $\mathsf{Ext}_{\mathsf{ks}}$ invokes $\mathsf{Ext}_{\mathsf{ss}}(\mathsf{lp},\mathbf{tr})$, obtaining $G(\mathbb{w})$. Finally, we run $\mathcal{D}$ on $\mathcal{A}_{\mathsf{ss}}$'s output $\mathbf{tr}$.

Fix lp, $\mathcal{R}_{\mathfrak{J}}$, $(C,\mathbb{x})$, and st. Consider an implicit Boolean matrix $\boldsymbol{M}$ (that depends on lp, $\mathcal{R}_{\mathfrak{J}}$, $\mathcal{A}_{\mathsf{ks}}$, and st), such that $M_{\varrho r} = 1$ iff the Π verifier accepts $\mathcal{A}_2$'s output given a fixed constraint ϱ and CRS generator's randomness r. Since $\varepsilon_1(\lambda)$ is non-negligible, the row $\varrho = 0$ of $\boldsymbol{M}$ has a non-negligible fraction of ones.

Since $\varepsilon_1(\lambda)$ is non-negligible and $\varepsilon'(\lambda)$ is negligible, $\varepsilon_1(\lambda) - \varepsilon'(\lambda)$ is non-negligible. Thus, each row of $\boldsymbol{M}$ has a non-negligible fraction $\varepsilon^*(\lambda) \in [\varepsilon_1(\lambda) - \varepsilon'(\lambda), \varepsilon_1(\lambda) + \varepsilon'(\lambda)]$ of ones. Hence, $\mathcal{A}_{\mathsf{ss}}$ always returns an admissible $\mathbf{tr}$. Moreover, $\mathcal{A}_{\mathsf{ss}}$ works in the expected time $n/\varepsilon^*(\lambda) \cdot \mathsf{poly}(\lambda) = \mathsf{poly}(\lambda)$.

Clearly, $\mathcal{D}$ succeeds with the same probability in both games. The second part (V accepts $\Rightarrow (\mathbb{x},\mathbb{w}) \notin \mathcal{R}_{\mathfrak{J}}$) holds iff $\mathsf{Ext}_{\mathsf{ss}}$ fails to extract a valid witness $\mathbb{w}$, i.e., with the probability $\mathsf{Adv}^{\mathrm{specsound}}_{\mathsf{Pgen},G,\Pi,n,\mathsf{Ext}_{\mathsf{ss}},\mathcal{A}_{\mathsf{ss}}}(\lambda)$. Hence,

$$\mathsf{Adv}^{\mathrm{bbks}}_{\mathsf{Pgen},G,\Pi,\mathsf{Ext}_{\mathsf{ks}},\mathcal{A}_{\mathsf{ks}}}(\lambda) = |\varepsilon_2(\lambda) - \varepsilon_1(\lambda)| \le \mathsf{Adv}^{\mathrm{specsound}}_{\mathsf{Pgen},G,\Pi,n,\mathsf{Ext}_{\mathsf{ss}},\mathcal{A}_{\mathsf{ss}}}(\lambda) \ .$$

Since $\mathsf{Ext}_{\mathsf{ks}}$ aborts with probability $1 - \varepsilon_1(\lambda)$, it works in the expected time $\leq n \cdot \varepsilon_1(\lambda)/(\varepsilon_1(\lambda) - \varepsilon'(\lambda)) \cdot \mathsf{poly}(\lambda) \approx n \cdot \mathsf{poly}(\lambda) = \mathsf{poly}(\lambda)$. □

E Punic's Security Proofs

E.1 Proof of Theorem 2 (Punic Is Complete And Zero-Knowledge)

Proof. **(1: perfect completeness.)** Since BLS is perfectly complete, we have only to check that the last verification equation in Fig. 5 holds. But

$$[\mathsf{a}(x)]_1 \bullet [\mathsf{b}(x)]_2 - [\mathsf{c}(x)]_1 \bullet [1]_2 = [\mathsf{h}(x)]_1 \bullet [Z_{\mathbb{H}}(x)]_2$$

due to the definition of $\mathsf{h}(X)$.

(2: perfect zero-knowledge.) We show that the simulator in Fig. 5 functions properly when $\varrho = 0$. Since $\boldsymbol{E}_1^0$ and $\boldsymbol{E}_2^0$ are zero matrices the almost everywhere perfectly-hiding property of FSE_γ (see Fig. 3) means FSE_γ is perfectly hiding. We perfectly simulate $[\mathsf{d}(x)]_1$ and $[\mathsf{e}(x)]_2$ by committing to $\mathbf{0}$. In the honest argument, $[\mathsf{a}(x)]_1$, $[\mathsf{b}(x)]_2$, and $[\mathsf{c}(x)]_1$ are uniformly random and independently distributed. Sim picks $r_\mathsf{a}, r_\mathsf{b}, r_\mathsf{c} \leftarrow\!\!\$\, \mathbb{F}$ and defines $[\mathsf{a}(x)]_1 \leftarrow r_\mathsf{a}[Z_{\mathbb{H}}(x)]_1$, $[\mathsf{b}(x)]_2 \leftarrow r_\mathsf{b}[Z_{\mathbb{H}}(x)]_2$, and $[\mathsf{c}(x)]_1 \leftarrow r_\mathsf{c}[Z_{\mathbb{H}}(x)]_1$ as VCF_γ-commitments to $\mathbf{0}$. (That is, it simulates the vector commitments.) Sim satisfies the verification equation by setting

$$[\mathsf{h}(x)]_1 \leftarrow [(\mathsf{a}(x)\mathsf{b}(x) - \mathsf{c}(x))/Z_{\mathbb{H}}(x)]_1 = r_\mathsf{a} r_\mathsf{b}[Z_{\mathbb{H}}(x)]_1 - r_\mathsf{c}[1]_1 \ .$$

Finally, BLS.π can be perfectly simulated (see [27]) using BLS.td. □

E.2 Proof of Theorem 3 (Punic is CRS-Indistinguishable)

Proof (Sketch). We can write this as a hybrid argument, where we first change $\boldsymbol{E}_1$ and then change $\boldsymbol{E}_2$. For simplicity, consider only the case when we change $\boldsymbol{E}_1$. $\mathcal{B}_1$ checks if $\mathcal{A}$'s output π^ϱ is an acceptable argument. If it is, $\mathcal{B}_1$ guesses that $\beta = 1$. Due to semi-adaptivity, the statement $([C]_1, \mathbb{x})$ does not change from game to game; moreover, since $\mathcal{A}$ succeeded, $([C]_1, \mathbb{x})$ does not belong to the language. Hence, $\mathcal{A}$'s succeeds in cheating iff the verification accepts. Moreover, one can have a single adversary $\mathcal{B}_1$ that does not depend on ϱ. □

E.3 Theorem 4 (Punic is Special-Sound)

Proof. Let $\mathcal{A}_{\mathsf{ss}}$ be a special soundness adversary that succeeds with probability ε. Recall the semi-adaptive computational (n, G)-special soundness game from Definition 7. In this game, the challenger forwards lp to $\mathcal{A}_{\mathsf{ss}}$, who replies with $\mathbf{tr}$. After that, the challenger invokes a special soundness extractor $\mathsf{Ext}_{\mathsf{ss}}$ who returns $G(\mathbb{w})$.

Fig. 12 depicts the special soundness extractor $\mathsf{Ext}_{\mathsf{ss}}$ that is given $\mathbf{tr}$ as an input. $\mathsf{Ext}_{\mathsf{ss}}$ loops over all constraints ϱ. For each ϱ, since FSE_1 is somewhere $[\cdot]_1$-extractable and $\mathrm{FSE}_1.\mathsf{ek}^\varrho$ is part of td^ϱ and thus tr^ϱ, $\mathsf{Ext}_{\mathsf{ss}}$ can use $\mathrm{FSE}_1.\mathsf{swExt}$

$\mathsf{Ext}_{\mathsf{ss}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}}, \mathbf{tr} = \{\mathsf{tr}^{\varrho} = ([C]_1, \mathbb{x}, \mathsf{crs}^{\varrho}, \mathsf{td}^{\varrho}, \pi^{\varrho})\}_{\varrho=1}^{n})$

1 : **for** $\varrho \in [1, n]$ **do** // π^{ϱ} contains $[\mathsf{d}^{\varrho}]_1$, td^{ϱ} contains $\mathrm{FSE}_1.\mathsf{ek}^{\varrho}$
2 : $\quad (G(\boldsymbol{\eta}^{\varrho}|_{N(\varrho)})^{\intercal}, [(\boldsymbol{\varphi}^{\varrho}|_{N(\varrho)})^{\intercal}, \varphi_{\mathsf{a}}^{\varrho}, \varphi_{\mathsf{c}}^{\varrho}]_1^{\intercal}) \leftarrow \mathrm{FSE}_1.\mathsf{swExt}\left(\mathrm{FSE}_1.\mathsf{ek}^{\varrho}, [\mathsf{d}^{\varrho}]_1\right)$;
3 : $\quad G(\mathbb{p}^{\varrho}) \leftarrow G(\boldsymbol{\eta}^{\varrho}|_{N(\varrho)})$;
4 : **endfor**
5 : $G(\mathbb{z}) \leftarrow \left(\begin{smallmatrix} G(\mathbb{x}) \\ \bot \end{smallmatrix}\right)$; // Glue partial witnesses together:
6 : **for** $\varrho \in [1, n]$ **do for** $j \in N(\varrho)$ **do**
7 : $\quad$ **if** $G(\mathbb{z}_j) = \bot$ **then** $G(\mathbb{z}_j) \leftarrow G(\mathbb{p}_j^{\varrho})$;
8 : $\quad$ **elseif** $G(\mathbb{z}_j) \neq G(\mathbb{p}_j^{\varrho})$ **then return** $G(\mathbb{z}) \leftarrow \bot$; **fi**
9 : **endfor endfor**
10 : **return** $G(\mathbb{w})$; // $= G((\mathbb{z}_i)_{i=m_{\mathbb{x}}+1}^{m})$

Fig. 12. The semi-adaptive computational (n, G)-special soundness extractor $\mathsf{Ext}_{\mathsf{ss}}$ in Theorem 4.

to black-box extract a partial witness $G(\mathbb{p}^{\varrho})$ together with related values. After the loop finishes, $\mathsf{Ext}_{\mathsf{ss}}$ glues $\{G(\mathbb{p}^{\varrho})\}$ together to a full witness $G(\mathbb{w}) = [\mathbb{w}y]_1$ that satisfies all constraints.

The gluing process in Fig. 12 is a greedy algorithm that assigns coefficients of $[\mathbb{z}]_1$ by using partial witnesses one by one until it detects an inconsistency. At that point, $\mathsf{Ext}_{\mathsf{ss}}$ has found two inconsistent partial witnesses, and it aborts.

Let us analyze the probability that $\mathcal{A}_{\mathsf{ss}}$ succeeds (thus, V accepts all proofs π^{ϱ}), but the black-box special soundness extractor $\mathsf{Ext}_{\mathsf{ss}}$ aborts. In this case, either

(1) for some ϱ, $(\mathbb{x}, \mathbb{p}^{\varrho}) \notin \mathcal{R}_{\mathsf{loc},\mathfrak{J}}^{\varrho}$, that is, $\mathbb{p}^{\varrho}$ is either not consistent with the input $\mathbb{x}$ or does not satisfy the ϱth constraint (then $\mathsf{Ext}_{\mathsf{ss}}$ does not abort, but the verifier will not accept $\mathbb{w}$), or
(2) $(\mathbb{x}, \mathbb{p}^{\varrho}) \in \mathcal{R}_{\mathsf{loc},\mathfrak{J}}^{\varrho}$ for all ϱ, but there exist i, j such that $\mathsf{Cons}(\mathbb{p}^i, \mathbb{p}^j) = 0$ (then, the extractor aborts on step 8).

More precisely, $\mathcal{A}_{\mathsf{ss}}$ succeeds if at least one of the following holds:

(a) there exists a ϱ, such that

$$(\exists j \in ([1, m_{\mathbb{x}}] \cap N(\varrho)))\, \eta_j \neq \mathbb{x}_j \ ,$$

(b) there exists a ϱ, such that the black-box extracted partial witness $G(\mathbb{p}^{\varrho}) = G(\boldsymbol{\eta}^{\varrho}|_{N(\varrho)})$ does not satisfy the ϱth constraint, that is,

$$\left(\textstyle\sum_{j \in N(\varrho)} U_{\varrho j} \eta_j^{\varrho}\right) \cdot \left(\textstyle\sum_{j \in N(\varrho)} V_{\varrho j} \eta_j^{\varrho}\right) \neq \left(\textstyle\sum_{j \in N(\varrho)} W_{\varrho j} \eta_j^{\varrho}\right) \ ,$$

(c) there exist i and j, such that $\mathsf{Cons}(\mathbb{p}^i, \mathbb{p}^j) = 0$.

We construct three reductions that collectively succeed whenever $\mathcal{A}_{\mathsf{ss}}$ succeeds but $\mathsf{Ext}_{\mathsf{ss}}$ does not (either aborts or returns a wrong witness). Each reduction succeeds when the variable ev takes a specific value. Two of the reductions ($\mathcal{B}_{\mathsf{bls}}$ when $\mathsf{ev} = \mathsf{ev}_{\mathsf{bls}}$ and $\mathcal{B}_{\mathsf{qal}}$ when $\mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}$) are related to (but not equal to) reductions in [18,19,52], while the third one ($\mathcal{B}_{\mathsf{fposb}}$ when $\mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}$) is novel.

Moreover, instead of just $\mathsf{ev} = \mathsf{ev}_{\mathsf{bls}}$, we consider events $\mathsf{ev}^{\varrho}_{\mathsf{bls}}$, for $\varrho \in [1, n]$. Events $\mathsf{ev}^{i}_{\mathsf{bls}}$ and $\mathsf{ev}^{j}_{\mathsf{bls}}$ for $i \neq j$ can hold simultaneously. We will explain this now case-by-case.

$\underline{\mathsf{ev} = \mathsf{ev}^{\varrho}_{\mathsf{bls}},\ \varrho \in [1, n]:}$ in this case,

$$\mathrm{BLS}.\mathbb{x}^{\varrho} = (C^*, \mathsf{a}^{\varrho}, \mathsf{c}^{\varrho}, \mathsf{d}^{\varrho}, \mathsf{b}^{\varrho}, \mathsf{e}^{\varrho})^{\intercal} \notin \mathrm{colspace}\left(\begin{smallmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \end{smallmatrix}\right) \ ,$$

that is, Eq. (5) does not hold for this ϱ. Equivalently, there does not exist

$$\mathrm{BLS}.\mathbb{w}^{\varrho} = (\mathbb{z}^{\varrho}, r^{\varrho}_{C}, r^{\varrho}_{\mathsf{a}}, r^{\varrho}_{\mathsf{b}}, r^{\varrho}_{\mathsf{c}}, r^{\varrho}_{\mathsf{d}}, r^{\varrho}_{\mathsf{e}})^{\intercal} \ ,$$

such that Eq. (6) holds for ϱ.

In the contrary, if $\mathsf{ev} \notin \{\mathsf{ev}^{\varrho}_{\mathsf{bls}}\}$, then for each ϱ, there exists *at least* one $\mathbb{z}^{\varrho}$, such that $[C, \mathsf{d}^{\varrho}]_1$ and $[\mathsf{e}^{\varrho}]_2$ commit to $\mathbb{z}^{\varrho}$, while $[\mathsf{a}^{\varrho}]_1$, $[\mathsf{b}^{\varrho}]_2$, and $[\mathsf{c}^{\varrho}]_1$ commit to $\boldsymbol{U}\mathbb{z}^{\varrho}$, $\boldsymbol{V}\mathbb{z}^{\varrho}$, and $\boldsymbol{W}\mathbb{z}^{\varrho}$. Since V accepts, $(\forall \varrho)\mathsf{a}^{\varrho}\mathsf{b}^{\varrho} = \mathsf{c}^{\varrho}$. Thus, for

$$G(\eta^{\varrho}_{\mathsf{a}}) := \sum_{j \in N(\varrho)} U_{\varrho j} G(\eta^{\varrho}_{j})$$

and

$$G(\eta^{\varrho}_{\mathsf{c}}) := \sum_{j \in N(\varrho)} W_{\varrho j} G(\eta^{\varrho}_{j}) \ ,$$

we get

$$(\forall \varrho)\,(\exists \mathbb{z}^{\varrho})\, G(\eta^{\varrho}_{\mathsf{a}}) = G(\boldsymbol{U}_{\varrho}\mathbb{z}^{\varrho}) \wedge G_2(\eta^{\varrho}_{\mathsf{b}}) = G_2(\boldsymbol{V}_{\varrho}\mathbb{z}^{\varrho}) \wedge G(\eta^{\varrho}_{\mathsf{c}}) = G(\boldsymbol{W}_{\varrho}\mathbb{z}^{\varrho}) \ , \quad (10)$$

where $G_2(s) := [sy]_2$. Moreover,

$$[C^*(x)]_1 = \sum_{j=m_{\mathbb{x}}+1}^{n} \mathbb{z}^{\varrho}_j [\ell_j(x)]_1 + r^{\varrho}_{C} [Z_{\mathbb{H}}(x)]_1$$

and thus

$$[C]_1 = \sum_{j=1}^{m_{\mathbb{x}}} \mathbb{x}_j [\ell_j(x)]_1 + \sum_{j=m_{\mathbb{x}}+1}^{n} \mathbb{z}^{\varrho}_j [\ell_j(x)]_1 + r_C [Z_{\mathbb{H}}(x)]_1$$

uses the correct $\mathrm{R1CS}_{\mathsf{f}}$ statement $\mathbb{x}$. Thus, the cheating avenue a is impossible, and in the following cases, one only has to deal with the cheating strategies b and c.

$\underline{\mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}:}$ Eq. (5) (that is, $\mathrm{BLS}.\mathbb{x}^{\varrho} \in \mathrm{colspace}\left(\begin{smallmatrix} \boldsymbol{M}_1 \\ \boldsymbol{M}_2 \end{smallmatrix}\right)$) holds for all ϱ. In particular, Eq. (10) holds. However, there is no full witness $\mathbb{w}$, such that $\mathbb{z} = \left(\begin{smallmatrix} \mathbb{x} \\ \mathbb{w} \end{smallmatrix}\right)$ satisfies all constraints:

$$(\neg \exists \mathbb{z} \forall \varrho)\,(G(\eta^{\varrho}_{\mathsf{a}}) = G(\boldsymbol{U}_{\varrho}\mathbb{z}) \ \wedge \ G_2(\eta^{\varrho}_{\mathsf{b}}) = G_2(\boldsymbol{V}_{\varrho}\mathbb{z}) \ \wedge \ G(\eta^{\varrho}_{\mathsf{c}}) = G(\boldsymbol{W}_{\varrho}\mathbb{z})) \ . \quad (11)$$

Due to Remark 1, this corresponds to the cheating avenue c.

$\mathcal{B}^{\varrho}_{\mathsf{bls}}(\mathsf{p}, \mathrm{BLS.lp} = [\boldsymbol{M}]_*, \mathrm{BLS.lt} = \left(\begin{smallmatrix}\boldsymbol{M}_1\\ \boldsymbol{M}_2\end{smallmatrix}\right), \mathrm{BLS.crs})$

Compute x from $Z_{\mathbb{H}}(x) \in \mathbb{F}$ that is given in $\boldsymbol{M}_1$; $y \leftarrow\!\!\$\ \mathbb{F}^*$;
$\mathsf{lp} \leftarrow \mathrm{VCF}_1.\mathsf{ck} = ([(x^j)_{j=0}^n, y]_1, [(x^j)_{j=0}^n, y]_2)$;
$\mathbf{tr} \leftarrow \mathcal{A}_{\mathsf{ss}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}})$; // $\mathbb{x} = (\mathbb{z}_1, \ldots, \mathbb{z}_{m_{\mathbb{x}}})$
if $\mathbf{tr}$ is not admissible **then return** $\bot$;
// Use only single transcript tr^{ϱ}; $\pi^{\varrho} = \left(\left[\mathsf{a}^{\varrho}, \mathsf{c}^{\varrho}, \mathsf{d}^{\varrho}, \mathsf{h}^{\varrho}\right]_1, \left[\mathsf{b}^{\varrho}, \mathsf{e}^{\varrho}\right]_2, \mathrm{BLS}.\pi^{\varrho}\right)$
$[C^*(x)]_1 \leftarrow [C]_1 - \sum_{j=1}^{m_{\mathbb{x}}} \mathbb{x}_j [\ell_j(x)]_1$;
$\mathrm{BLS}.\mathbb{x} \leftarrow ([C^*(x), \mathsf{a}^{\varrho}, \mathsf{c}^{\varrho}, (\mathsf{d}^{\varrho})^{\mathsf{T}}]_1, [\mathsf{b}^{\varrho}, (\mathsf{e}^{\varrho})^{\mathsf{T}}]_2)^{\mathsf{T}}$;
return $(\mathrm{BLS}.\mathbb{x}, \mathrm{BLS}.\pi)$;

Fig. 13. The BLS strong-soundness adversary $\mathcal{B}_{\mathsf{bls}}$.

$\underline{\mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}}$*:* Eq. (5) (that is, $\mathrm{BLS}.\mathbb{x}^{\varrho} \in \mathrm{colspace}(\left(\begin{smallmatrix}\boldsymbol{M}_1\\ \boldsymbol{M}_2\end{smallmatrix}\right))$ holds for all ϱ. Moreover, a full witness $\mathbb{w}$ exists, such that $\mathbb{z} = \left(\begin{smallmatrix}\mathbb{x}\\ \mathbb{w}\end{smallmatrix}\right)$ satisfies all constraints; that is, Eq. (11) does not hold. In this case, only the cheating avenue b is possible. Thus, there must exists a ϱ, such that $(\boldsymbol{U}\mathbb{z})_{\varrho}(\boldsymbol{V}\mathbb{z})_{\varrho} \neq (\boldsymbol{W}\mathbb{z})_{\varrho}$.

In Lemmas 5 to 7 (see Appendices E.4 to E.6), we construct reductions $\mathcal{B}^{\varrho}_{\mathsf{bls}}$ (for $\varrho \in [1, n]$), $\mathcal{B}_{\mathsf{fposb}}$, $\mathcal{B}_{\mathsf{qal}}$, such that

$$\begin{aligned}
\Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} | \mathsf{ev} = \mathsf{ev}^{\varrho}_{\mathsf{bls}}] &\leq \mathsf{Adv}^{\mathrm{strsound}}_{\mathsf{Pgen}, \mathcal{D}^{\varrho}_{\mathrm{par}}, \mathrm{BLS}, \mathcal{B}^{\varrho}_{\mathsf{bls}}}(\lambda) \ , \\
\Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}] &\leq \mathsf{Adv}^{\mathrm{fposb}}_{\mathsf{Pgen}, [\cdot]_1, n, \mathrm{VCF}_1, \mathcal{B}_{\mathsf{fposb}}}(\lambda) \ , \\
\Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}] &\leq \mathsf{Adv}^{\mathrm{qalinres}}_{\mathsf{Pgen}, n, \mathcal{B}_{\mathsf{qal}}}(\lambda) \ .
\end{aligned}$$

The three reductions emulate $\mathsf{Ext}_{\mathsf{ss}}$ from Fig. 12 internally; each takes care of one possibility when $\mathsf{Ext}_{\mathsf{ss}}$ can fail. Clearly, if $\mathcal{A}_{\mathsf{ss}}$ succeeds and none of the cases $\mathsf{ev} = \mathsf{ev}^{\varrho}_{\mathsf{bls}}$, $\mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}$, $\mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}$ is true, then $\mathsf{Ext}_{\mathsf{ss}}$ succeeds. Assuming Lemmas 5 to 7, Theorem 4 holds. □

E.4 Lemma 5

Lemma 5. *Let n be the number of R1CS$_{\mathfrak{f}}$ constraints and $\varrho \in [1, n]$. Assume* BLS *is quasi-adaptively strongly sound. For every PPT special soundness adversary $\mathcal{A}_{\mathsf{ss}}$, there exists a PPT $\mathcal{B}^{\varrho}_{\mathsf{bls}}$, such that*

$$\Pr[\mathcal{A}_{\mathsf{ss}} \ \textit{succeeds} | \mathsf{ev} = \mathsf{ev}^{\varrho}_{\mathsf{bls}}] \leq \mathsf{Adv}^{\mathrm{strsound}}_{\mathsf{Pgen}, \mathcal{D}^{\varrho}_{\mathrm{par}}, \mathrm{BLS}, \mathcal{B}^{\varrho}_{\mathsf{bls}}}(\lambda) \ .$$

Here, a part of the reduction's input is $[\boldsymbol{M}]_*$, which depends on $\mathrm{FSE}_1.\mathsf{ck}$ and $\mathrm{FSE}_2.\mathsf{ck}$. Thus, one cannot reprogram $[\boldsymbol{M}]_*$ inside the reduction. This is why we construct a different reduction $\mathcal{B}^{\varrho}_{\mathsf{bls}}$ for each $\varrho \in [1, n]$ that works in the case the commitments corresponding to the ϱth CRS are not properly formed. Since the distribution of $\mathrm{BLS.lp}$ depends on ϱ, $\mathcal{B}^{\varrho}_{\mathsf{bls}}$ is only required to be secure for the distribution of $\mathrm{BLS.lp}$ corresponding to ϱ.

Proof. Let $\mathcal{A}_{\mathsf{ss}}$ be a semi-adaptive computational $(n, [\cdot]_1)$-special soundness adversary against Punic. Assume $\mathsf{ev} = \mathsf{ev}_{\mathsf{bls}}^{\varrho}$ for some ϱ. That is, $\mathcal{A}_{\mathsf{ss}}$ makes the verifier to accept but there does not exist

$$\mathrm{BLS}.\mathbb{w}^{\varrho} = (\mathbb{z}^{\varrho}, r_C^{\varrho}, r_{\mathsf{a}}^{\varrho}, r_{\mathsf{b}}^{\varrho}, r_{\mathsf{c}}^{\varrho}, r_{\mathsf{d}}^{\varrho}, r_{\mathsf{e}}^{\varrho})^{\intercal} \ ,$$

such that Eq. (6) holds.

We start the reduction by creating lp for $\mathcal{A}_{\mathsf{ss}}$. For this, we need to know x. We black-box extract x form $Z_{\mathbb{H}}(x)$, which is an entry of $\boldsymbol{M}_1$. We can do that since BLS is strongly-sound and thus $\mathcal{B}$ has access to $\mathrm{BLS}.\mathsf{lt} = \left(\begin{smallmatrix}\boldsymbol{M}_1\\ \boldsymbol{M}_2\end{smallmatrix}\right)$. We then obtain $\mathbf{tr} = \{\mathsf{tr}^{\varrho} = ([C]_1, \mathbb{x}, \mathsf{crs}^{\varrho}, \mathsf{td}^{\varrho}, \pi^{\varrho})\}$ from $\mathcal{A}_{\mathsf{ss}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}})$.

After that, $\mathcal{B}_{\mathsf{bls}}$ emulates $\mathsf{Ext}_{\mathsf{ss}}$ from Fig. 12, but does not execute things not needed for this concrete reduction to work, and adds things to finish the reduction. In particular, since $\mathcal{B}_{\mathsf{bls}}$ only deals with one constraint ϱ, we do not have to loop over all constraints. Hence, $\mathcal{B}_{\mathsf{bls}}^{\varrho}$ uses its input $\mathrm{BLS}.\mathsf{crs}$ as the BLS CRS corresponding to the ϱth constraint. $\mathcal{B}_{\mathsf{bls}}$ obtains from $\mathbf{tr}$ an argument π^{ϱ} and returns a BLS output computed from the rest of $\mathbf{tr}$. See Fig. 13.

Clearly,

$$\Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{bls}}^{\varrho}] \leq \mathsf{Adv}^{\mathrm{strsound}}_{\mathsf{Pgen}, \mathcal{D}_{\mathrm{par}}^{\varrho}, \mathrm{BLS}, \mathcal{B}_{\mathsf{bls}}^{\varrho}}(\lambda) \ .$$

□

E.5 Lemma 6

Lemma 6 (Position-Binding). *Let n be the number of R1CS$_{\mathsf{f}}$ constraints. Assume* FSE_1 *is somewhere $[\cdot]_1$-extractable and* VCF_1 *is $[\cdot]_1$-position-binding. For every PPT special soundness adversary $\mathcal{A}_{\mathsf{ss}}$, there exists a PPT $\mathcal{B}_{\mathsf{fposb}}$, such that*

$$\Pr[\mathcal{A}_{\mathsf{ss}} \textit{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}] \leq \mathsf{Adv}^{\mathrm{fposb}}_{\mathsf{Pgen}, [\cdot]_1, n, \mathrm{VCF}_1, \mathcal{B}_{\mathsf{fposb}}}(\lambda) \ .$$

Proof. Assume $\mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}$. Let $\mathcal{A}_{\mathsf{ss}}$ be a PPT special soundness adversary that succeeds with probability

$$\varepsilon := \Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} = 1 | \mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}] \ .$$

In Fig. 14, we depict a $[\cdot]_1$-position-binding adversary $\mathcal{B}_{\mathsf{fposb}}$. $\mathcal{B}_{\mathsf{fposb}}(\mathsf{p}, \mathrm{VCF}_1.\mathsf{ck})$ constructs Punic's lp and then calls $\mathcal{A}_{\mathsf{ss}}(\mathsf{lp}, \mathcal{R}_{\mathfrak{J}})$ to obtain $\mathbf{tr}$.

After that, $\mathcal{B}_{\mathsf{fposb}}$ loops over all constraints $\varrho \in [1, n]$. On the ϱth step, $\mathcal{B}_{\mathsf{fposb}}$ does the following:

1. Since FSE_1 is somewhere $[\cdot]_1$-extractable and $\mathrm{FSE}_1.\mathsf{ek}^{\varrho} \in \mathsf{td}^{\varrho}$, $\mathcal{B}_{\mathsf{fposb}}$ can use the somewhere extraction procedure of FSE. More precisely, $\mathcal{B}_{\mathsf{fposb}}$ black-box extracts
$$\begin{pmatrix} G(\boldsymbol{\eta}^{\varrho}) \\ \begin{bmatrix} \boldsymbol{\varphi}^{\varrho} \\ \varphi_{\mathsf{a}}^{\varrho} \\ \varphi_{\mathsf{c}}^{\varrho} \end{bmatrix}_1 \end{pmatrix} \leftarrow [\boldsymbol{E}_1^{\varrho}]_1 \cdot \begin{pmatrix} \mathbb{z}^{\varrho} \\ r_{\mathsf{a}}^{\varrho} \\ r_{\mathsf{c}}^{\varrho} \\ r_C^{\varrho} \end{pmatrix}$$

$\mathcal{B}_{\mathsf{fposb}}(\mathsf{p}, \mathrm{VCF}_1.\mathsf{ck} = ([(x^j)_{j=0}^n, y]_1, [(x^j)_{j=0}^n, y]_2))$ // $\mathrm{VCF}_1.\mathsf{ck} = \mathrm{VCF}_2.\mathsf{ck}$

$\mathsf{lp} \leftarrow \mathrm{VCF}_1.\mathsf{ck}$;
tr $\leftarrow \mathcal{A}_{\mathsf{ss}}(\mathsf{lp})$;
if tr is not admissible **then return** $\bot$;
for $\varrho \in [1, n]$ **do** // π^ϱ contains $[\mathsf{d}^\varrho]_1$, td^ϱ contains $\mathrm{FSE}_1.\mathsf{ek}^\varrho$
$(G(\boldsymbol{\eta}^\varrho|_{N(\varrho)})^\intercal, [(\boldsymbol{\varphi}^\varrho|_{N(\varrho)})^\intercal, \varphi_{\mathsf{a}}^\varrho, \varphi_{\mathsf{c}}^\varrho]_1^\intercal) \leftarrow \mathrm{FSE}_1.\mathsf{swExt}\left(\mathrm{FSE}_1.\mathsf{ek}^\varrho, [\mathsf{d}^\varrho]_1\right)$;
endfor
// Find a pair of inconsistent partial witnesses
for $i \in [1, n]$ **do**
for $j \in [i+1, n]$ **do**
for $k \in N(i) \cap N(j)$ **do** // Partial witnesses are different
if $G(\eta_k^i) \neq G(\eta_k^j)$ **then return** $([C]_1, k, G(\eta_k^i), [\varphi_k^i]_1, G(\eta_k^j), [\varphi_k^j]_1)$; **fi**
endfor
endfor

Fig. 14. The $[\cdot]_1$-position-binding adversary $\mathcal{B}_{\mathsf{fposb}}$ in Lemma 6.

as in Eq. (8), where

$$G(\eta_{\beta_j}^\varrho) = \textstyle\sum_k \mathbb{z}_k^\varrho G(e_{\beta_j}^k) = G(\mathbb{z}_{\beta_j}^\varrho) \ ,$$
$$[\varphi_{\beta_j}^\varrho]_1 = \textstyle\sum_k \mathbb{z}_k^\varrho [Q_{\ell_k,\beta_j}(x)]_1 + r_C^\varrho \cdot [Q_{Z_{\mathbb{H}},\beta_j}(x)]_1 \ ,$$

and thus

$$(G(\eta_{\beta_j}^\varrho), [\varphi_{\beta_j}^\varrho]_1) = \mathsf{LOpen}(\mathrm{VCF}_1.\mathsf{ck}, [C]_1, \beta_j, (\mathbb{z}^\varrho, r_C^\varrho)) \ .$$

The last equality follows from taking the quotients of the equation

$$(\ldots, \textstyle\sum_k \mathbb{z}_k^\varrho [\ell_k(x)]_1 + r_C^\varrho \cdot [Z_{\mathbb{H}}(x)]_1) = \mathsf{Com}(\mathrm{VCF}_1.\mathsf{ck}, \mathbb{z}; r_C) \ .$$

(We actually only need to extract $(G(\eta_{\beta_j}^\varrho), [\varphi_{\beta_j}^\varrho]_1)$ for all $\mathfrak{f}$ bits $\beta_j \in N(\varrho)$.) Now, recall we are in the case $\mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}$. Thus, for each ϱ, there exists a full witness $\mathbb{z}^\varrho$ that agrees with the ϱth black-box extracted value, but there is no full witness $\mathbb{z}$ that agrees with the extracted values of all n constraints. By Remark 1, there exist $i \neq j$, such that $\mathbb{z}^i$ satisfies the ith constraint and $\mathbb{z}^j$ satisfies the jth constraint, but $\mathbb{z}^i$ does not satisfy the jth constraint. Hence, there exists a coefficient $k \in N(i) \cap N(j)$, such that the kth coefficients of $\mathbb{z}^i$ and $\mathbb{z}^j$ are different. During the ith and the jth iteration of the loop, we black-box extract $G(\eta_k^i) = G(\mathbb{z}_k^i)$ and $G(\eta_k^j) = G(\mathbb{z}_k^j)$, together with local proofs certifying that $[\mathbb{z}_k^i]_1$ and $[\mathbb{z}_k^j]_1$ are both valid local openings of $[C]_1$. $\mathcal{B}_{\mathsf{fposb}}$ breaks the $[\cdot]_1$-position-binding of VCF_1 by returning $G(\eta_k^i) \neq G(\eta_k^j)$ with local proofs.

Hence,

$$\Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{fposb}}] \leq \mathsf{Adv}^{\mathrm{fposb}}_{\mathsf{Pgen},[\cdot]_1,n,\mathrm{VCF}_1,\mathcal{B}_{\mathsf{fposb}}}(\lambda) \ .$$

□

$\mathcal{B}_{\mathsf{qal}}(\mathsf{p}, \mathrm{VCF}_1.\mathsf{ck} = ([(x^j)_{j=0}^n, y]_1, [(x^j)_{j=0}^n, y]_2))$ // $\mathrm{VCF}_1.\mathsf{ck} = \mathrm{VCF}_2.\mathsf{ck}$

$\mathsf{lp} \leftarrow \mathrm{VCF}_1.\mathsf{ck}$;
$\mathbf{tr} \leftarrow \mathcal{A}_{\mathsf{ss}}(\mathsf{lp})$;
if $\mathbf{tr}$ is not admissible **then return** $\bot$;
for $\varrho \in [1, n]$ **do**
  // π^ϱ contains $([\mathsf{a}^\varrho, \mathsf{c}^\varrho, \mathsf{d}^\varrho, \mathsf{h}^\varrho]_1, [\mathsf{b}^\varrho, \mathsf{e}^\varrho]_2)$, td^ϱ contains $(\mathrm{FSE}_1.\mathsf{ek}^\varrho, \mathrm{FSE}_2.\mathsf{ek}^\varrho)$
  $(G(\boldsymbol{\eta}^\varrho|_{N(\varrho)}), [\boldsymbol{\varphi}^\varrho|_{N(\varrho)}, \varphi_{\mathsf{a}}^\varrho, \varphi_{\mathsf{c}}^\varrho]_1^\intercal) \leftarrow \mathrm{FSE}_1.\mathsf{swExt}\left(\mathrm{FSE}_1.\mathsf{ek}^\varrho, [\mathsf{d}^\varrho]_1\right)$;
  $G(\eta_{\mathsf{a}}^\varrho) \leftarrow \sum_{j \in N(\varrho)} U_{\varrho j} G(\eta_j^\varrho)$; $G(\eta_{\mathsf{c}}^\varrho) \leftarrow \sum_{j \in N(\varrho)} W_{\varrho j} G(\eta_j^\varrho)$;
  $(G_2(\eta_{\mathsf{b}}^\varrho), [\varphi_{\mathsf{b}}^\varrho]_2) \leftarrow \mathrm{FSE}_2.\mathsf{swExt}\left(\mathrm{FSE}_2.\mathsf{ek}^\varrho, [\mathsf{e}^\varrho]_2\right)$;
  if $G(\eta_{\mathsf{a}}^\varrho) \bullet G_2(\eta_{\mathsf{b}}^\varrho) \neq G(\eta_{\mathsf{c}}^\varrho) \bullet G_2(1)$ **then**
    return $(\varrho, [\mathsf{a}^\varrho]_1, G(\eta_{\mathsf{a}}^\varrho), [\varphi_{\mathsf{a}}^\varrho, \mathsf{c}^\varrho]_1, G(\eta_{\mathsf{c}}^\varrho), [\varphi_{\mathsf{c}}^\varrho, \mathsf{h}^\varrho]_1, [\mathsf{b}^\varrho]_2, G_2(\eta_{\mathsf{b}}^\varrho), [\varphi_{\mathsf{b}}^\varrho]_2)$; **fi**
endfor

Fig. 15. The QALINRES adversary $\mathcal{B}_{\mathsf{qal}}$ in Lemma 7.

E.6 Lemma 7

Lemma 7 (QALINRES **reduction).** *Let n be the number of R1CS$_{\mathsf{f}}$ constraints. Assume* FSE_1 *is somewhere $[\cdot]_1$-extractable,* FSE_2 *is somewhere $[\cdot]_2$-extractable, and* QALINRES *holds. For every PPT special soundness adversary $\mathcal{A}_{\mathsf{ss}}$, there exists a PPT $\mathcal{B}_{\mathsf{qal}}$, such that*

$$\Pr[\mathcal{A}_{\mathsf{ss}} \textit{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}] \leq \mathsf{Adv}^{\mathrm{qalinres}}_{\mathsf{Pgen}, n, \mathcal{B}_{\mathsf{qal}}}(\lambda) \ .$$

Proof. Let $\mathcal{A}_{\mathsf{ss}}$ be a special soundness adversary. Assume $\mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}$, that is, there exists a full witness $\mathbb{w}$ that is consistent with all openings. In this case, the adversary's only cheating avenue is to leave some constraint, say, constraint ϱ, unsatisfied. Then, the local opening of constraint ϱ shows inconsistency.

In Fig. 15, we depict a QALINRES adversary (see Definition 1). Here, $G_2(s) := [sy]_2$. $\mathcal{B}_{\mathsf{qal}}$ starts by creating lp and obtaining $\mathbf{tr}$ from $\mathcal{A}_{\mathsf{ss}}$. After that, $\mathcal{B}_{\mathsf{qal}}$ loops over all the constraints. After black-box extracting the partial witness of a constraint ϱ, $\mathcal{B}_{\mathsf{qal}}$ checks that this constraint is satisfied by the partial witness.

More precisely, since FSE_1 is somewhere $[\cdot]_1$-extractable and FSE_2 is somewhere $[\cdot]_2$-extractable, $\mathcal{B}_{\mathsf{qal}}$ black-box extracts

1.
$$\begin{pmatrix} G(\boldsymbol{\eta}^\varrho) \\ \begin{bmatrix} \boldsymbol{\varphi}^\varrho \\ \varphi_{\mathsf{a}}^\varrho \\ \varphi_{\mathsf{c}}^\varrho \end{bmatrix}_1 \end{pmatrix} \leftarrow [\boldsymbol{E}_1^\varrho]_1 \cdot \begin{pmatrix} \mathbb{z}^\varrho \\ r_{\mathsf{a}}^\varrho \\ r_C^\varrho \\ r_{\mathsf{c}}^\varrho \end{pmatrix}$$

as in Eq. (8), where say

$$[\varphi_{\mathsf{a}}^\varrho]_1 = \left[\sum_k \mathbb{z}_k^\varrho Q_{u_k, \varrho} + r_{\mathsf{a}}^\varrho Q_{Z_{\mathbb{H}}, \varrho}(x) \right]_1 \ .$$

Setting

$$G(\eta_{\mathsf{a}}^\varrho) \leftarrow \sum U_{\varrho j} G(\mathbb{z}_j^\varrho) = G((\boldsymbol{U}\mathbb{z}^\varrho)_\varrho) \ ,$$

we get

$$(G(\eta_{\mathsf{a}}^{\varrho}), [\varphi_{\mathsf{a}}^{\varrho}]_1) = \mathsf{LOpen}(\mathrm{VCF}_1.\mathsf{ck}, [\mathsf{a}^{\varrho}]_1, \varrho, (\boldsymbol{U}\mathbb{z}^{\varrho}, r_{\mathsf{a}}^{\varrho})) \ .$$

2. $\begin{pmatrix} G_2(\eta_{\mathsf{b}}^{\varrho}) \\ [\varphi_{\mathsf{b}}^{\varrho}]_2 \end{pmatrix} \leftarrow [\boldsymbol{E}_2^{\varrho}]_2 \cdot \begin{pmatrix} \mathbb{z}^{\varrho} \\ r_{\mathsf{b}}^{\varrho} \end{pmatrix}$, where

$$G_2(\eta_{\mathsf{b}}^{\varrho}) = \sum V_{\varrho j} G_2(\mathbb{z}_j^{\varrho}) = G_2((\boldsymbol{V}\mathbb{z}^{\varrho})_{\varrho})$$

and

$$\begin{aligned}[\varphi_{\mathsf{b}}^{\varrho}]_2 &= \sum_k \mathbb{z}_k^{\varrho} [Q_{v_k,\varrho}(x)]_2 + r_{\mathsf{b}}^{\varrho} [Q_{Z_{\mathbb{H}},\varrho}(x)]_2 \\ &= \mathsf{LOpen}(\mathrm{VCF}_2.\mathsf{ck}, [\mathsf{b}^{\varrho}(x)]_2, \varrho, (\boldsymbol{V}\mathbb{z}^{\varrho}, r_{\mathsf{b}}^{\varrho})) \ .\end{aligned}$$

(In this reduction, we only need to extract $(G(\eta_{\mathsf{a}}^{\varrho}), [\varphi_{\mathsf{a}}^{\varrho}]_1, G(\eta_{\mathsf{c}}^{\varrho}), [\varphi_{\mathsf{c}}^{\varrho}]_1)$ and $(G_2(\eta_{\mathsf{b}}^{\varrho}), [\varphi_{\mathsf{b}}^{\varrho}]_2)$.)

$\mathcal{B}_{\mathsf{qal}}$ checks if

$$G(\eta_{\mathsf{a}}^{\varrho}) \bullet G_2(\eta_{\mathsf{b}}^{\varrho}) = G(\eta_{\mathsf{c}}^{\varrho}) \bullet G_2(1) \ .$$

If the check fails (the constraint is unsatisfied), it uses the partial witness to break the QALINRES assumption. Since we are in the case $\mathsf{bad} = \mathsf{ev}_{\mathsf{qalinres}}$, $\mathcal{B}_{\mathsf{qal}}$ must succeed for at least one constraint ϱ.

Hence,

$$\Pr[\mathcal{A}_{\mathsf{ss}} \text{ succeeds} | \mathsf{ev} = \mathsf{ev}_{\mathsf{qalinres}}] \le \mathsf{Adv}^{\mathrm{qalinres}}_{\mathsf{Pgen},n,\mathcal{B}_{\mathsf{qal}}}(\lambda) \ .$$

□