Bulletproofs for R1CS: Bridging the Completeness-Soundness Gap and a ZK Extension

Gil Segev*

Abstract

Bulletproofs, introduced by Bünz, Bootle, Boneh, Poelstra, Wuille and Maxwell (IEEE S&P, 2018), is a highly efficient non-interactive argument system that does not require a trusted setup. Recently, Bünz (PhD Thesis, 2023) extended Bulletproofs to support arguments for rank-1 constraint satisfaction (R1CS) systems, a widely-used representation for arithmetic satisfiability problems. Although the argument system constructed by Bünz preserves the attractive properties of Bulletproofs, it presents a gap between its completeness and soundness guarantees: The system is complete for a restricted set of instances, but sound only for a significantly broader set. Although argument systems for such gap relations nevertheless provide clear and concrete guarantees, the gaps they introduce may lead to various inconsistencies or undesirable gaps within proofs of security, especially when used as building blocks within larger systems.

In this work we show that the argument system presented by Bünz can be extended to bridge the gap between its completeness and soundness, and to additionally provide honest-verifier zero-knowledge. For the extended argument system, we introduce a refined R1CS relation that captures the precise set of instances for which both completeness and soundness hold without resorting to a gap formulation. The extended argument system preserves the performance guarantees of the argument system presented by Bünz, and yields a non-interactive argument system using the Fiat-Shamir transform.

^{*}Hebrew University of Jerusalem and Coinbase. Work done while at Coinbase.

Contents

1	Introduction	1
2	Preliminaries2.1 Interactive Argument Systems2.2 Inner-Product Arguments	
3	The R1CS* Relation	5
4	An R1CS* ZK-Argument System 4.1 Completeness	6 9 10 11
Re	eferences	16
\mathbf{A}	From mIP Arguments to IP Arguments	17
В	The Chung et al. mIP Argument System	19
\mathbf{C}	An Auxiliary Claim	28

Introduction

Bulletproofs is a practical argument system constructed by Bünz, Bootle, Boneh, Poelstra, Wuille and Maxwell [BBB+18], building on the techniques of Bootle, Cerulli, Chaidos, Groth and Petit [BCC+16]. The Bulletproofs argument system does not require a trusted setup, and provides logarithmic-length inner-product, range and arithmetic circuit satisfiability arguments relative to a committed witness. The practical applicability of Bulletproofs was further demonstrated by Bünz [Bün23], who constructed a Bulletproofs argument system for rank-1 constraint satisfaction (R1CS) systems relative to a committed witness. Such systems generalize arithmetic circuits, and have become the interface to a variety of state-of-the-art argument systems (see, for example, [GGP+13, Gro16, BCR+19, CHM+20, OB22] and the references therein). Specifically, an R1CS instance is parameterized by integers $m, n \ge 1$ and a prime q, and consists of three matrices $A, B, C \in \mathbb{Z}_q^{m \times n}$. In turn, an R1CS witness is vector $z \in \mathbb{Z}_q^n$ that satisfies $Az \circ Bz = Cz$, where \circ denotes the Hadamard (element-wise) product.¹

Bulletproofs for R1CS. The argument system constructed by Bünz [Bün23] enables to prove that a given R1CS system is satisfiable by a witness such that a commitment to a part (or to all) of the witness is provided together with the R1CS instance. Specifically, Bünz constructed logarithmic-length arguments for the following relation \mathcal{R}_{R1CS} which is defined with respect to parameters $m, r, n \in \mathbb{N}$, where $m \geq 1$ and $1 \leq r \leq n$, and a cyclic group \mathbb{G} of prime order q with generators $G, H \in \mathbb{G}^{n+m}$. The relation \mathcal{R}_{R1CS} consists of all pairs $((T, A, B, C), (\boldsymbol{x}, \boldsymbol{y}))$, where

$$T \in \mathbb{G}, \quad A, B, C \in \mathbb{Z}_q^{m \times n}, \quad \boldsymbol{x} \in \mathbb{Z}_q^r, \quad \boldsymbol{y} \in \mathbb{Z}_q^{n-r},$$

which satisfy the following two requirements²:

- 1. $T = \langle ((\boldsymbol{x}||0^{n-r}) || 0^m), \boldsymbol{G} \rangle$.
- 2. $Az \circ Bz = Cz$ for $z = (x||y) \in \mathbb{Z}_q^n$.

That is, the group element T is a commitment to x, and the matrices A, and B and C define an R1CS system that is satisfied by z = (x||y). In his description of the relation \mathcal{R}_{R1CS} , Bünz allows the group element T to be of the more general form

$$T = \langle ((\boldsymbol{x}||\boldsymbol{u}) || 0^m), \boldsymbol{G} \rangle + \langle (\boldsymbol{v} || 0^m), \boldsymbol{H} \rangle$$

for some $u \in \mathbb{Z}_q^{n-r}$ and $v \in \mathbb{Z}_q^n$ which, as noted by Bünz, can be set to the all-zero vectors. However, his argument system does not provide completeness for such instances (i.e., when u and v may be arbitrary vectors instead of the all-zero vectors). In addition, when analyzing the soundness of his argument system, Bünz presented an algorithm that extracts a witness z = (x||y) for which $Az \circ Bz = Cz$ as required, but for which T is of the even more general form

$$T = \langle ((\boldsymbol{x}||\boldsymbol{t_2}) \mid\mid \boldsymbol{t_3}), \boldsymbol{G} \rangle + \langle ((\boldsymbol{t_1'}||\boldsymbol{t_2'}) \mid\mid \boldsymbol{t_3'}), \boldsymbol{H} \rangle$$

for some $t_1' \in \mathbb{Z}_q^r$, $t_2, t_2' \in \mathbb{Z}_q^{n-r}$ and $t_3' \in \mathbb{Z}_q^m$. As the vectors t_3 and t_3' may not be the all-zero vectors, this is not a valid witness even with respect to instances of the general form T = $\langle ((\boldsymbol{x}||\boldsymbol{u}) \mid | 0^m), \boldsymbol{G} \rangle + \langle (\boldsymbol{v} \mid | 0^m), \boldsymbol{H} \rangle$. In fact, as we demonstrate in Section 4, there are instances in

For any $\ell \geq 1$, $\boldsymbol{x} = (x_1, \dots, x_\ell) \in \mathbb{Z}_q^{\ell}$ and $\boldsymbol{y} = (y_1, \dots, y_\ell) \in \mathbb{Z}_q^{\ell}$, the Hadamard product $\boldsymbol{x} \circ \boldsymbol{y} \in \mathbb{Z}_q^{\ell}$ is defined as $\boldsymbol{x} \circ \boldsymbol{y} = (x_1 \cdot y_1, \dots, x_\ell \cdot y_\ell) \in \mathbb{Z}_q^{\ell}$.

For any $\ell \geq 1$, $\boldsymbol{c} = (c_1, \dots, c_\ell) \in \mathbb{Z}_q^{\ell}$ and $\boldsymbol{G} = (G_1, \dots, G_\ell) \in \mathbb{G}^{\ell}$, we let $\langle \boldsymbol{c}, \boldsymbol{G} \rangle = \sum_{i=1}^{\ell} c_i \cdot G_i \in \mathbb{G}$.

which t_3 and t'_3 are not the all-zero vectors, and for which an efficient prover can always convince the verifier to accept. Thus, this gap cannot be bridged by only refining the analysis provided by Bünz.

The gap between the completeness and soundness in the argument system presented by Bünz can be formalized by viewing his argument system as an argument system for a "gap" relation: Given an "outer" relation \mathcal{R}_{out} and an "inner" relation $\mathcal{R}_{in} \subsetneq \mathcal{R}_{out}$, the honest prover can convince the verifier to accept any instance x when provided with an inner-witness w such that $(x, w) \in \mathcal{R}_{in}$, whereas any efficient malicious prover that convinces the verifier to accept an instance x with a non-negligible probability can be used to extract an outer-witness w such that $(x, w) \in \mathcal{R}_{out}$. That is, completeness is provided for instances of the inner relation \mathcal{R}_{in} , whereas soundness is guaranteed for instances of the outer relation \mathcal{R}_{out} . Although argument systems for such gap relations nevertheless provide clear and concrete guarantees, the gaps they introduce may lead to various inconsistencies or undesirable gaps within proofs of security, especially when used as building blocks within larger systems.

Our contributions. In this work we show that the argument system presented by Bünz can be extended to bridge the above-discussed gap between its completeness and soundness, and to additionally provide honest-verifier zero-knowledge. Specifically, we introduce a relation \mathcal{R}_{R1CS^*} that lies in between the outer and inner relations resulting from the analysis presented by Bünz, and extend his argument system to provide completeness, soundness and honest-verifier zero-knowledge for \mathcal{R}_{R1CS^*} without considering gap relations. The extended argument system preserves the performance guarantees of the argument system presented by Bünz, where for instances in which $T = \langle ((\boldsymbol{x}||0^{n-r}) \mid| 0^m), \boldsymbol{G} \rangle$ the two argument systems coincide. Furthermore, the extended argument system is public coin and yields a non-interactive argument system using the Fiat-Shamir transform [FS87, AFK23].

2 Preliminaries

Let $\mathbb G$ be a cyclic group of prime order q that is generated by $G \in \mathbb G$. Throughout this document we denote scalars via lower-case letters (e.g., $x \in \mathbb Z_q$), vectors of scalars via boldface lower-case letters (e.g., $x = (x_1, \dots, x_\ell) \in \mathbb Z_q^\ell$), group elements via upper-case letters (e.g., $G \in \mathbb G$), and vectors of group elements via boldface upper-case letters (e.g., $G = (G_1, \dots, G_\ell) \in \mathbb G^\ell$). For $x = (x_1, \dots, x_\ell) \in \mathbb Z_q^\ell$, $y = (y_1, \dots, y_\ell) \in \mathbb Z_q^\ell$ and $G = (G_1, \dots, G_\ell) \in \mathbb G^\ell$, we let $\langle x, G \rangle = \sum_{i=1}^\ell x_i \cdot G_i \in \mathbb G$ and $x \circ y = (x_1 \cdot y_1, \dots, x_\ell \cdot y_\ell) \in \mathbb Z_q^\ell$. For $x \in \mathbb Z_q^*$ and integer $\ell \geq 1$ we let $x^\ell = (x, \dots, x^\ell) \in \mathbb Z_q^\ell$.

2.1 Interactive Argument Systems

An interactive argument system for a relation $\mathcal{R} = \{\mathcal{R}_{\sigma}\}_{\sigma \in \{0,1\}^*}$ is a triplet $\Pi = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ of probabilistic polynomial-time algorithms. The common-reference string generation algorithm \mathcal{K} receives as input the unary representation 1^{κ} of the security parameter $\kappa \in \mathbb{N}$, and outputs a common-reference string σ . For any $\kappa \in \mathbb{N}$ and for any σ produced by $\mathcal{K}(1^{\kappa})$, the prover algorithm \mathcal{P} and the verifier algorithm \mathcal{V} define an interactive protocol $\langle \mathcal{P}(\sigma,\cdot,\cdot),\mathcal{V}(\sigma,\cdot)\rangle$. The input of the prover consists of a common-reference string σ , an instance x and a witness w, and the input of the verifier consists of the common-reference string σ and the instance x. We assume without loss of generality that any common-reference string σ produced by $\mathcal{K}(1^{\kappa})$ is of length of least κ bits (thus, we do not need to provide \mathcal{P} and \mathcal{V} with 1^{κ} as part of their input). We denote by $\operatorname{tr} \leftarrow \langle \mathcal{P}(\sigma, x, w), \mathcal{V}(\sigma, x) \rangle$ the probabilistic process of producing a transcript of the protocol, and denote by $\operatorname{Out}_{\mathcal{V}}\langle \mathcal{P}(\sigma, x, w), \mathcal{V}(\sigma, x) \rangle$ the random variable corresponding to the output of the verifier, in both cases when the prover and verifier follow the instructions of the protocol.

In what follows we present the standard notions of completeness, honest-verifier zero-knowledge and witness-extended emulation for interactive argument systems.

Definition 2.1 (Perfect Completeness). An interactive argument system $\Pi = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ for a relation $\mathcal{R} = \{\mathcal{R}_{\sigma}\}_{\sigma \in \{0,1\}^*}$ has perfect completeness if for any algorithm \mathcal{A} and for any $\kappa \in \mathbb{N}$ it holds that

$$\Pr\left[(x,w) \notin \mathcal{R}_{\sigma} \text{ or } \mathsf{Out}_{\mathcal{V}} \langle \mathcal{P}(\sigma,x,w), \mathcal{V}(\sigma,x) \rangle = 1 \,\middle|\, \begin{matrix} \sigma \leftarrow \mathcal{K}(1^{\kappa}) \\ (x,w) \leftarrow \mathcal{A}(\sigma) \end{matrix}\right] = 1.$$

Definition 2.2 (Public Coin). An interactive argument system $\Pi = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ is *public coin* if all messages sent by the honest verifier are uniformly distributed and independent of the honest verifier's input and of all messages previously sent by the prover.

Definition 2.3 (Perfect Special Honest-Verifier Zero-Knowledge). An interactive argument system $\Pi = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ for a relation $\mathcal{R} = \{\mathcal{R}_{\sigma}\}_{\sigma \in \{0,1\}^*}$ has perfect special honest-verifier zero-knowledge if there exists a probabilistic polynomial-time simulator \mathcal{S} such that for any algorithms \mathcal{A}_1 and \mathcal{A}_2 it holds that

$$\Pr \begin{bmatrix} (x, w) \in \mathcal{R}_{\sigma} & \sigma \leftarrow \mathcal{K}(1^{\kappa}), (x, w, \rho) \leftarrow \mathcal{A}_{1}(\sigma) \\ and & \mathcal{A}_{2}(\mathsf{tr}) = 1 & \mathsf{tr} \leftarrow \langle \mathcal{P}(\sigma, x, w), \mathcal{V}(\sigma, x; \rho) \rangle \end{bmatrix}$$

$$= \Pr \begin{bmatrix} (x, w) \in \mathcal{R}_{\sigma} & \sigma \leftarrow \mathcal{K}(1^{\kappa}), (x, w, \rho) \leftarrow \mathcal{A}_{1}(\sigma) \\ and & \mathcal{A}_{2}(\mathsf{tr}) = 1 & \mathsf{tr} \leftarrow \mathcal{S}(\sigma, x, \rho) \end{bmatrix}$$

for all $\kappa \in \mathbb{N}$, where $\rho \in \{0,1\}^*$ denotes the randomness of the verifier \mathcal{V} .

Definition 2.4 (Witness-Extended Emulation [Lin03]). An interactive argument system $\Pi = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ for a relation $\mathcal{R} = \{\mathcal{R}_{\sigma}\}_{\sigma \in \{0,1\}^*}$ has statistical witness-extended emulation if for any algorithm \mathcal{P}^* there exists an expected polynomial-time emulator \mathcal{E} such that for any algorithms \mathcal{A}_1 and \mathcal{A}_2 there exists a negligible function $\nu(\cdot)$ such that

$$\begin{vmatrix} \Pr \left[\mathcal{A}_{2}(\mathsf{tr}) = 1 \middle| \begin{matrix} \sigma \leftarrow \mathcal{K}(1^{\kappa}), (x, u) \leftarrow \mathcal{A}_{1}(\sigma) \\ \mathsf{tr} \leftarrow \langle \mathcal{P}^{*}(\sigma, x, u), \mathcal{V}(\sigma, x) \rangle \end{matrix} \right] \\ - \Pr \left[\begin{matrix} \mathcal{A}_{2}(\mathsf{tr}) = 1 \ and \\ \mathsf{tr} \ is \ accepting \implies (x, w) \in \mathcal{R}_{\sigma} \end{matrix} \middle| \begin{matrix} \sigma \leftarrow \mathcal{K}(1^{\kappa}), (x, u) \leftarrow \mathcal{A}_{1}(\sigma) \\ (\mathsf{tr}, w) \leftarrow \mathcal{E}^{\mathcal{O}}(\sigma, x) \end{matrix} \right] \middle| \leq \nu(\kappa) \end{vmatrix}$$

for all $\kappa \in \mathbb{N}$, where the oracle $\mathcal{O} = \langle \mathcal{P}^*(\sigma, x, u), \mathcal{V}(\sigma, x) \rangle$ permits rewinding to a specific point and resuming with fresh randomness for the verifier. Such an argument system has *computational* witness-extended emulation if the above holds when restricting \mathcal{A}_1 , \mathcal{A}_2 and \mathcal{P}^* to probabilistic polynomial time.

In order to prove that the argument systems we present provide witness-extended emulation, we rely on the general forking lemma of Bootle, Cerulli, Chaidos, Groth and Petit [BCC⁺16] that we now state by following their presentation. Let Π be a $(2\mu + 1)$ -move argument system with μ challenges c_1, \ldots, c_{μ} . Let $n_1, \ldots, n_{\mu} \geq 1$ and consider $\Pi_{i=1}^{\mu} n_i$ accepting transcripts in the following tree structure: The tree is labeled with a statement x, it has depth μ and $\Pi_{i=1}^{\mu} n_i$ leaves, where each node at depth i has n_i children labeled with distinct values for the ith challenge c_i . We refer to such transcripts as an (n_1, \ldots, n_{μ}) -tree of accepting transcripts. For simplicity, in the following lemma Bootle et al. assumed that the challenges are chosen uniformly from \mathbb{Z}_q , for a λ -bit prime q such that $\lambda = \Omega(\kappa)$, where κ is the security parameter (we note that the lemma holds also when some or all of the challenges are chosen uniformly from \mathbb{Z}_q^*).

Lemma 2.5 ([BCC⁺16]). Let $\mu = \mu(\kappa)$ be a function of the security parameter $\kappa \in \mathbb{N}$, let Π be a $(2\mu + 1)$ -move public-coin argument system for a relation \mathcal{R} , and for each $i \in [\mu]$ let $n_i = n_i(\kappa) \geq 1$ such that $\Pi_{i=1}^{\mu} n_i$ is upper bounded by a polynomial in κ . If there exists a probabilistic polynomial-time algorithm that always succeeds in extracting a valid witness from any (n_1, \ldots, n_{μ}) -tree of accepting transcripts, then Π has statistical witness-extended emulation.

2.2 Inner-Product Arguments

As a building block for constructing an R1CS argument system, following Bünz [Bün23] we rely on an argument system for the inner-product relation \mathcal{R}_{IP} defined as follows:

The Relation $\mathcal{R}_{\mathrm{IP}}$

Let $d \ge 1$ be an integer, and let \mathbb{G} be a cyclic group of prime order q. The relation \mathcal{R}_{IP} consists of all pairs $((G, H, G, H, P, \omega), (u, v, \alpha))$, where

$$G, H \in \mathbb{G}^d$$
, $G, H, P \in \mathbb{G}$, $u, v \in \mathbb{Z}_q^d$, $\omega, \alpha \in \mathbb{Z}_q$,

which satisfy the following requirements:

1.
$$P = \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \alpha \cdot H$$
.

2.
$$\omega = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$$
.

The relation \mathcal{R}_{IP} slightly differs from the inner-product relation considered by Bünz [Bün23] and by Bünz, Bootle, Boneh, Poelstra, Wuille and Maxwell [BBB+18], which did not allow the group element P to contain the additional term $\alpha \cdot H$ (in our setting this additional term is used for additionally providing honest-verifier zero-knowledge). Nevertheless, we show that the approach of Bünz et al. for constructing an argument system for the inner-product relation they considered applies also to the more subtle relation \mathcal{R}_{IP} . Specifically, we show that an argument system for the above-defined relation \mathcal{R}_{IP} can be obtained from an argument system for the following "multiplicative" inner-product relation \mathcal{R}_{mIP} that is defined as follows:

The Relation \mathcal{R}_{mIP}

Let $d \geq 1$ be an integer, and let \mathbb{G} be a cyclic group of prime order q. The relation \mathcal{R}_{mIP} consists of all pairs $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha))$, where

$$m{G}, m{H} \in \mathbb{G}^d, \quad G, H, P \in \mathbb{G}, \qquad m{u}, m{v} \in \mathbb{Z}_q^d, \quad lpha \in \mathbb{Z}_q,$$

which satisfy

$$P = \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot G + \alpha \cdot H .$$

We construct an argument system for the relation \mathcal{R}_{IP} based on the argument system constructed by Chung, Han, Ju, Kim and Seo [CHJ⁺22] for the relation \mathcal{R}_{mIP} (we describe their argument system in full detail in Appendix B).³ Specifically, given a probabilistic polynomial-time group-generation algorithm that on input the security parameter $\kappa \in \mathbb{N}$ produces a description of a cyclic group \mathbb{G} of a κ -bit prime order q and 2d+2 uniformly and independently sampled generators $G, H \in \mathbb{G}^d$ and $G, H \in \mathbb{G}$, we prove the following theorem in Appendix A:

Theorem 2.6. Let $t : \mathbb{N} \to \mathbb{N}$ be any function of the security parameter $\kappa \in \mathbb{N}$ such that $d = d(\kappa) = 2^{t(\kappa)}$ is polynomial. Assuming the hardness of the DL problem for expected polynomial-time

 $^{^{3}}$ The argument system of Chung et al. [CHJ $^{+}$ 22] in fact supports a more general, weighted, form of the multiplicative inner-product relation which is not needed for our purposes

algorithms, there exists an argument system Π_{IP} for the d-dimensional inner-product relation \mathcal{R}_{IP} that has perfect completeness, perfect special honest-verifier zero-knowledge, and computational witness-extended emulation. Furthermore, the argument system is public coin, and the prover communicates $2 \cdot \log_2 d + 2$ group elements and 3 field elements.

As noted in Section 2.1, in order to prove that the R1CS* argument system we present provides witness-extended emulation, we rely on the general forking lemma of Bootle, Cerulli, Chaidos, Groth and Petit [BCC+16] (see Lemma 2.5 above). For this purpose, we rely on the following lemma that we prove in Appendix A for establishing the witness-extended emulation property of the argument system $\Pi_{\rm IP}$:

Lemma 2.7. There exists a probabilistic polynomial-time algorithm Ext that, on input any \mathcal{R}_{IP} instance (G, H, G, H, P, ω) together with any corresponding (2, 4, ..., 4, 5)-transcript tree of depth $\log_2 d + 2$ for the argument system Π_{IP} , where $d = 2^t \geq 1$, produces either a witness $(\boldsymbol{u}, \boldsymbol{v}, \alpha)$ such that $((G, H, G, H, P, \omega), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{IP}$ or a non-trivial discrete-logarithm relation for (G, H, G, H).

In fact, for proving Lemma 2.7 we rely on the following minor refinement of the notion of a transcript tree. Recall that each node at depth i of a transcript tree has n_i children labeled with distinct values for the ith challenge $c_i \in \mathbb{Z}_q$. For proving Lemma 2.7 we require that not for any two children $j \neq j' \in [n_i]$ of a node at depth i it holds that $c_{i,j} \neq c_{i,j'}$, but also $c_{i,j} \neq -c_{i,j'}$. Since we consider the general forking lemma of Bootle et al. for a λ -bit prime q such that $\lambda = \Omega(\kappa)$, where κ is the security parameter, their lemma still holds for this minor refinement.

3 The R1CS* Relation

In this section we introduce the \mathcal{R}_{R1CS^*} relation which, as discussed above, lies in between the outer and inner relations resulting from the analysis presented by Bünz. The \mathcal{R}_{R1CS^*} relation is defined as follows:

The Relation $\mathcal{R}_{\text{R1CS}^*}$

Let $m, r, n \in \mathbb{N}$ be such that $m \geq 1$ and $1 \leq r \leq n$, let \mathbb{G} be a cyclic group of prime order q, and let $G, H \in \mathbb{G}^{n+m}$ and $G, H \in \mathbb{G}$ be 2(n+m)+2 generators. The relation \mathcal{R}_{R1CS^*} consists of all pairs $((T, A, B, C), (\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta))$, where

$$T \in \mathbb{G}, \quad A, B, C \in \mathbb{Z}_q^{m \times n}, \quad \boldsymbol{x}, \boldsymbol{x'} \in \mathbb{Z}_q^r, \quad \boldsymbol{y}, \boldsymbol{y'} \in \mathbb{Z}_q^{n-r}, \quad \eta \in \mathbb{Z}_q$$

which satisfy the following requirements for $z=(x||y)\in\mathbb{Z}_q^n$ and $z'=(x'||y')\in\mathbb{Z}_q^n$:

- 1. $T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) || A\boldsymbol{z'}), \boldsymbol{G} \rangle + \langle (0^n || B\boldsymbol{z'}), \boldsymbol{H} \rangle + \eta \cdot H.$
- 2. $Az \circ Bz = Cz$.
- 3. $Az' \circ Bz' = 0^m$.
- 4. $Az \circ Bz' + Bz \circ Az' = Cz'$.
- 5. $A_{[1:r]}\boldsymbol{x'} = B_{[1:r]}\boldsymbol{x'} = C_{[1:r]}\boldsymbol{x'} = 0^m$, where $A_{[1:r]}, B_{[1:r]}, C_{[1:r]} \in \mathbb{Z}^{m \times r}$ denote the leftmost r columns of the matrices A, B and C, respectively.

The relation \mathcal{R}_{R1CS^*} differs from the relation considered by Bünz in two aspects. First, the relation \mathcal{R}_{R1CS^*} considers an additional generator H which is used both for providing honest-verifier zero-knowledge and for supporting instances in which T already consists of an additional randomizing element $\eta \cdot H$ (where η is made part of the witness).

Second, when comparing the relation $\mathcal{R}_{\text{R1CS}^*}$ to the outer and inner relations resulting from the analysis presented by Bünz (while ignoring the additional randomizing element $\eta \cdot H$ for the purpose of this comparison), note that the requirement $T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) \mid\mid A\boldsymbol{z'}), \boldsymbol{G} \rangle + \langle (0^n \mid\mid B\boldsymbol{z'}), \boldsymbol{H} \rangle$ indeed lies between the requirement $T = \langle ((\boldsymbol{x}||0^{n-r}) \mid\mid 0^m), \boldsymbol{G} \rangle$ to which the argument system presented by Bünz provides completeness, and the requirement $T = \langle ((\boldsymbol{x}||\boldsymbol{t_2}) \mid\mid \boldsymbol{t_3}), \boldsymbol{G} \rangle + \langle ((\boldsymbol{t_1'}||\boldsymbol{t_2'}) \mid\mid \boldsymbol{t_3'}), \boldsymbol{H} \rangle$ to which it provides soundness. Furthermore, note that the additional requirements (i.e., requirements 3–5) are always satisfied for $\boldsymbol{z'} = 0^n$, and thus the relation $\mathcal{R}_{\text{R1CS}^*}$ indeed contains all instances to which the argument system presented by Bünz provides completeness. Thus, we expect that for most applications an argument system for the $\mathcal{R}_{\text{R1CS}^*}$ relation may be used directly for proving R1CS instances in which $T = \langle ((\boldsymbol{x}||0^{n-r}) \mid| 0^m), \boldsymbol{G} \rangle$.

If, for some applications, it is nevertheless required that soundness holds specifically for $T = \langle ((\boldsymbol{x}||0^{n-r}) \mid \mid 0^m), \boldsymbol{G} \rangle$, this can be resolved by additionally providing an argument of knowledge for the relation that consists of all instances $((G_1, \ldots G_r, T), \boldsymbol{x})$ where $T = \langle \boldsymbol{x}, (G_1, \ldots, G_r) \rangle$ and $\boldsymbol{x} \in \mathbb{Z}_q^{r,4}$ As shown by Attema and Cramer [AC20], such an argument can be provided either via a Σ -protocol in which the prover sends a single group element and r field elements, or by extending the Σ -protocol to a $\lceil \log_2(r+1) \rceil$ -round public-coin protocol in which the prover sends $2 \cdot \lceil \log_2(r+1) \rceil$ group elements and 3 field elements (thus, the classic Σ -protocol is a good practical fit for applications in which r is comparable to $\log_2(m+n)$). However, given that providing such an additional proof may in some cases significantly increase the overhead, it may be preferable to first examine whether the security analysis of application under consideration can be refined to rely on the relation \mathcal{R}_{R1CS^*} and thus avoid the additional proof.

4 An R1CS* ZK-Argument System

In this section we present and analyze an argument system Π_{R1CS^*} for the $\mathcal{R}_{\text{R1CS}^*}$ relation. Following the approach of Bünz, Bootle, Boneh, Poelstra, Wuille and Maxwell [BBB⁺18], the key idea observed by Bünz [Bün23] is that an R1CS instance (T, A, B, C) with respect to a given set of generators, where $T \in \mathbb{G}$ and $A, B, C \in \mathbb{Z}_q^{m \times n}$, can be transformed in a probabilistic manner into a single \mathcal{R}_{IP} instance with respect to a modified set of generators (see Section 2.2 for the definition of the relation \mathcal{R}_{IP}). That is, the m constraints defined by the instance (T, A, B, C) can combined into a single inner-product constraint.

Specifically, note that a solution $z \in \mathbb{Z}_q^n$ to the system $Az \circ Bz = Cz$ exists if and only if there exist vectors $\mathbf{w_A}, \mathbf{w_B} \in \mathbb{Z}_q^n$ such that: (1) $Az - \mathbf{w_A} = 0^m$, (2) $Bz - \mathbf{w_B} = 0^m$, and (3) $Cz - \mathbf{w_A} \circ \mathbf{w_B} = 0^m$. On the one hand, this increases the number of constraints from m to 3m. On the other hand, however, this "uncouples" the Hadamard product $Az \circ Bz$ involving the solution z. Such an uncoupling is crucial since the given commitment T commits to the individual entries of z and not to the products of each two individual entries of z. Now, for proving that $Az - \mathbf{w_A} = 0^m$, following an appropriate commitment from the prover, the verifier can uniformly sample a scalar $\alpha \leftarrow \mathbb{Z}_q^*$, and the inner-product argument system can be used for proving that $\langle \alpha^m, Az - \mathbf{w_A} \rangle = 0$, where $\alpha^m = (\alpha, \dots, \alpha^m) \in \mathbb{Z}_q^m$. If the prover successfully convinces the verifier with a non-negligible probability over the choice of α , then this holds for m+1 distinct values of α , and we are ensured that $Az - \mathbf{w_A} = 0^m$. The same approach can be taken for proving that $Bz - \mathbf{w_B} = 0^m$ and $Cz - \mathbf{w_A} \circ \mathbf{w_B} = 0^m$, resulting in three inner-product instances. For ensuring that the same $\mathbf{w_A}$ and $\mathbf{w_B}$ are used, and for further compressing the three inner-product instances into a single instance,

⁴Note that if $T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) \mid\mid A\boldsymbol{z'}), \boldsymbol{G} \rangle + \langle (0^n \mid\mid B\boldsymbol{z'}), \boldsymbol{H} \rangle + \eta \cdot H$ and $T = \langle ((\hat{\boldsymbol{x}}||0^{n-r}) \mid\mid 0^m), \boldsymbol{G} \rangle$ for $\boldsymbol{x} \neq \hat{\boldsymbol{x}}$, then this provides a non-trivial discrete-logarithm relation for $(\boldsymbol{G}, \boldsymbol{H}, H)$. Therefore, assuming the hardness of the DL problem for expected polynomial-time algorithms, the additional argument guarantees that $\boldsymbol{x} = \hat{\boldsymbol{x}}$.

the verifier samples $\alpha, \beta, \gamma \leftarrow \mathbb{Z}_q^*$, and the argument system includes a wide variety of technical complications for enabling the parties to essentially construct an inner-product instance roughly equivalent to the constraint

$$\langle \boldsymbol{\alpha}^m, A\boldsymbol{z} - \boldsymbol{w}_{\boldsymbol{A}} \rangle + \langle \boldsymbol{\beta}^m, B\boldsymbol{z} - \boldsymbol{w}_{\boldsymbol{B}} \rangle + \langle \boldsymbol{\gamma}^m, C\boldsymbol{z} - \boldsymbol{w}_{\boldsymbol{A}} \circ \boldsymbol{w}_{\boldsymbol{B}} \rangle = 0$$
.

In what follows we present the argument system, and then state and prove its completeness, soundness and zero-knowledge guarantees. The argument system Π_{R1CS^*} , which uses as a building block the inner-product argument system Π_{IP} provided by Theorem 2.6 (which we prove in Appendix A), is defined as follows (our modifications to the argument system presented by Bünz, which are discussed below, are colored red for convenience):

The Argument System Π_{R1CS^*}

• Public parameters:

- 1. Integers $m, r, n \in \mathbb{N}$ such that $m \geq 1, 1 \leq r \leq n$, and $n + m = 2^t$ for some integer $t \geq 1$.
- 2. Cyclic group \mathbb{G} of prime order q and 2(n+m)+2 generators $G, H \in \mathbb{G}^{n+m}$ and $G, H \in \mathbb{G}$.

• Inputs:

- 1. \mathcal{P} : Instance (T, A, B, C) and witness $(\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \boldsymbol{\eta}) \in \mathbb{Z}_q^r \times \mathbb{Z}_q^r \times \mathbb{Z}_q^{n-r} \times \mathbb{Z}_q^{n-r} \times \mathbb{Z}_q$.
- 2. V: Instance (T, A, B, C).

• Execution:

1. The prover \mathcal{P} samples $r \leftarrow \mathbb{Z}_q$, lets $z = (x||y) \in \mathbb{Z}_q^n$, computes

$$S = \langle ((\boldsymbol{x'}||\boldsymbol{y}) \mid |A\boldsymbol{z}), \boldsymbol{G} \rangle + \langle (0^n \mid |B\boldsymbol{z}), \boldsymbol{H} \rangle + \boldsymbol{r} \cdot \boldsymbol{H} \in \mathbb{G}$$

and sends S to the verifier \mathcal{V} .

- 2. The verifier \mathcal{V} samples $\alpha, \beta, \gamma, \delta \leftarrow \mathbb{Z}_q^*$ independently and uniformly, and sends $(\alpha, \beta, \gamma, \delta)$ to the prover \mathcal{P} .
- 3. Letting $G = (G_1, \ldots, G_{n+m})$, each party computes

$$\mu = \alpha \cdot \gamma \in \mathbb{Z}_{q}$$

$$\boldsymbol{\delta} = (\delta, \dots, \delta, 1^{n-r}) \in \mathbb{Z}_{q}^{n}$$

$$\boldsymbol{\delta}^{-1} = (\delta^{-1}, \dots, \delta^{-1}, 1^{n-r}) \in \mathbb{Z}_{q}^{n}$$

$$\boldsymbol{G'} = (G_{1}, \dots, G_{n}, \gamma^{-1} \cdot G_{n+1}, \dots, \gamma^{-m} \cdot G_{n+m}) \in \mathbb{G}^{n+m}$$

$$\boldsymbol{c} = \boldsymbol{\mu}^{m} A + \boldsymbol{\beta}^{m} B - \boldsymbol{\gamma}^{m} C \in \mathbb{Z}_{q}^{n}$$

$$\boldsymbol{\omega} = \langle \boldsymbol{\alpha}^{m}, \boldsymbol{\beta}^{m} \rangle + \delta^{2} \cdot \langle \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle \in \mathbb{Z}_{q}$$

$$P = \delta^{-1} \cdot T + S + \langle (\delta^{2} \cdot \boldsymbol{\alpha}^{n} \mid | -\boldsymbol{\beta}^{m}), \boldsymbol{G'} \rangle + \langle (\boldsymbol{c} \circ \boldsymbol{\delta} \mid | -\boldsymbol{\alpha}^{m}), \boldsymbol{H} \rangle \in \mathbb{G}$$

and the prover \mathcal{P} additionally lets $\mathbf{z'} = (\mathbf{x'}||\mathbf{y'}) \in \mathbb{Z}_q^n$ and computes

$$\begin{aligned} \boldsymbol{u} &= ((\boldsymbol{x'}||\boldsymbol{y}) + \delta^{-1} \cdot (\boldsymbol{x}||\boldsymbol{y'}) + \delta^2 \cdot \boldsymbol{\alpha}^n \mid \mid (A\boldsymbol{z} + \delta^{-1} \cdot A\boldsymbol{z'}) \circ \boldsymbol{\gamma}^m - \boldsymbol{\beta}^m) \in \mathbb{Z}_q^{n+m} \\ \boldsymbol{v} &= (\boldsymbol{c} \circ \boldsymbol{\delta} \mid \mid B\boldsymbol{z} - \boldsymbol{\alpha}^m + \delta^{-1} \cdot B\boldsymbol{z'}) \in \mathbb{Z}_q^{n+m} \\ \boldsymbol{\eta'} &= r + \delta^{-1} \cdot \boldsymbol{\eta} \end{aligned}$$

4. The parties invoke the inner-product argument Π_{IP} with the instance $(\mathbf{G'}, \mathbf{H}, G, H, P, \omega)$, where the prover \mathcal{P} takes the role of the prover using the witness $(\mathbf{u}, \mathbf{v}, \eta')$, and the verifier \mathcal{V} takes the role of the verifier and then outputs its output.

The argument system Π_{R1CS^*} is obtained from the argument system presented by Bünz by first introducing the additional vectors $\mathbf{x'}$ and $\mathbf{y'}$ for supporting the specific structure of the group element T as specified by the relation \mathcal{R}_{R1CS^*} , and by introducing the element $\eta \cdot H$ for randomizing the group element S in order to provide semi-honest zero-knowledge. Then, by additionally modifying the group element S and the vectors \mathbf{u} and \mathbf{v} , as described above in red font for supporting the additional vectors $\mathbf{x'}$ and $\mathbf{y'}$, this enables to provide both completeness and soundness for the relation \mathcal{R}_{R1CS^*} (setting $\mathbf{x'} = 0^r$, $\mathbf{y'} = 0^{n-r}$ and $\eta = 0$ yields the argument system presented by Bünz).

The following theorem captures the completeness, soundness, zero-knowledge and prover communication complexity of the argument system Π_{R1CS^*} . For formalizing the soundness of the argument system we assume that the cyclic group \mathbb{G} is produced by a group-generation algorithm, and that the generators G, H, G and H are uniformly and independently sampled. This enables us to prove that the argument system provides computational witness-extended emulation assuming the hardness of the DL problem for expected polynomial-time algorithms. Specifically, following Bootle et al. [BCC⁺16] and Bünz et al. [BBB⁺18], when including the description of the group and generators as part of the instance to the relation, we show that the argument system provides statistical witness-extended emulation for extracting either a valid witness or a non-trivial discrete-logarithm relation for (G, H, G, H).

Theorem 4.1. Let $t: \mathbb{N} \to \mathbb{N}$ be any function of the security parameter $\kappa \in \mathbb{N}$ such that $2^{t(\kappa)}$ is polynomial. Assuming the hardness of the DL problem for expected polynomial-time algorithms, then for any polynomials $m = m(\kappa)$, $r = r(\kappa)$ and $n = n(\kappa)$ such that $m \ge 1$, $1 \le r \le n$ and $n + m = 2^t$, it holds that Π_{R1CS^*} is an argument system for the relation \mathcal{R}_{R1CS^*} with perfect completeness, perfect special honest-verifier zero-knowledge and computational witness-extended emulation. Furthermore, the argument system is public coin, and the prover communicates $2 \cdot \log_2(n+m) + 3$ group elements and 3 field elements.

The proof of Theorem 4.1 is provided in Sections 4.1, 4.2 and 4.3 and which consider the completeness, zero-knowledge and witness-extended emulation properties, respectively. As for the communication complexity, note that the prover communicates a single group element, and then additionally communicates $2 \cdot \log_2(n+m) + 2$ group elements and 3 field elements when taking the role of the prover in the inner-product argument $\Pi_{\rm IP}$ (see Theorem 2.6 for the properties of the inner-product argument system $\Pi_{\rm IP}$).

Padding R1CS* instances. Note that we presented the argument system Π_{R1CS^*} for parameters n and m such that n+m is a power of 2, since we presented the argument system Π_{IP} for any dimension which is a power of 2. Dealing with the more general case in which n+m may not be a power of 2 can be done by padding any R1CS* instance (T, A, B, C) with "empty constraints". That is, by adding m' < n+m all-zero rows to the matrices $A, B, C \in \mathbb{Z}_q^{m \times n}$ to obtain matrices $A', B', C' \in \mathbb{Z}_q^{(m+m') \times n}$ for which n+m+m' is a power of 2. As a result, this requires including 2m' additional generators $G_{n+m+1}, H_{n+m+1}, \ldots, G_{n+m+m'}, H_{n+m+m'} \in \mathbb{G}$ in the public parameters.

additional generators $G_{n+m+1}, H_{n+m+1}, \ldots, G_{n+m+m'}, H_{n+m+m'} \in \mathbb{G}$ in the public parameters. We observe that for any $\boldsymbol{w} = (\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta) \in \mathbb{Z}_q^r \times \mathbb{Z}_q^r \times \mathbb{Z}_q^{n-r} \times \mathbb{Z}_q^{n-r} \times \mathbb{Z}_q$, it holds that \boldsymbol{w} is a valid witness for an instance (T, A, B, C) if and only if it is a valid witness for the padded instance (T, A', B', C'). Specifically, for $\boldsymbol{z} = (\boldsymbol{x}||\boldsymbol{y})$ and $\boldsymbol{z'} = (\boldsymbol{x'}||\boldsymbol{y'})$, it holds that $A'\boldsymbol{z'} = (A\boldsymbol{z'}||0^{m'})$ and

 $B'\mathbf{z'} = (B\mathbf{z'}||0^{m'})$, and therefore

$$T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) \mid| A\boldsymbol{z'}), \boldsymbol{G} \rangle + \langle (0^n \mid| B\boldsymbol{z'}), \boldsymbol{H} \rangle + \eta \cdot H$$

if and only if

$$T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) || A'\boldsymbol{z'}), (\boldsymbol{G}||G_{n+m+1}, \dots, G_{n+m+m'}) \rangle + \langle (0^n || B'\boldsymbol{z'}), (\boldsymbol{H}||H_{n+m+1}, \dots, H_{n+m+m'}) \rangle + \eta \cdot \boldsymbol{H}.$$

Similarly, it holds that:

- $Az \circ Bz = Cz$ if and only if $A'z \circ B'z = C'z$.
- $Az' \circ Bz' = 0^m$ if and only if $A'z' \circ B'z' = 0^{m+m'}$.
- $Az \circ Bz' + Bz \circ Az' = Cz'$ if and only if $A'z \circ B'z' + B'z \circ A'z' = C'z'$.
- $A_{[1:r]} x' = B_{[1:r]} x' = C_{[1:r]} x' = 0^m$ if and only if $A'_{[1:r]} x' = B'_{[1:r]} x' = C'_{[1:r]} x' = 0^{m+m'}$.

4.1 Completeness

We prove the following lemma by showing that the completeness of the argument system Π_{R1CS^*} is directly inherited from that of the inner-product argument system Π_{IP} .

Lemma 4.2. The argument system Π_{R1CS^*} has perfect completeness.

Proof. Let \mathbb{G} be a cyclic group of prime order q with 2(m+n)+2 generators $G, H \in \mathbb{G}^{n+m}$ and $G, H \in \mathbb{G}$, and let $((T, A, B, C), (\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta)) \in \mathcal{R}_{R1CS^*}$. We show that for the inner-product instance (G', H, G, H, P, ω) and witness $(\boldsymbol{u}, \boldsymbol{v}, \eta')$, as computed by the prover and verifier, it holds that $((G', H, G, H, P, \omega), (\boldsymbol{u}, \boldsymbol{v}, \eta')) \in \mathcal{R}_{IP}$. That is, we have to show that $\omega = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$ and that $P = \langle \boldsymbol{u}, \boldsymbol{G'} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \eta' \cdot H$. First, \boldsymbol{u} and \boldsymbol{v} are defined by the prover as

$$\boldsymbol{u} = ((\boldsymbol{x'}||\boldsymbol{y}) + \delta^{-1} \cdot (\boldsymbol{x}||\boldsymbol{y'}) + \delta^{2} \cdot \boldsymbol{\alpha}^{n} \mid | (A\boldsymbol{z} + \delta^{-1} \cdot A\boldsymbol{z'}) \circ \boldsymbol{\gamma}^{m} - \boldsymbol{\beta}^{m}) \in \mathbb{Z}_{q}^{n+m}$$
$$\boldsymbol{v} = (\boldsymbol{c} \circ \boldsymbol{\delta} \mid | B\boldsymbol{z} - \boldsymbol{\alpha}^{m} + \delta^{-1} \cdot B\boldsymbol{z'}) \in \mathbb{Z}_{q}^{n+m},$$

and therefore

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \langle (\boldsymbol{x'}||\boldsymbol{y}) + \delta^{-1} \cdot (\boldsymbol{x}||\boldsymbol{y'}) + \delta^{2} \cdot \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle + \langle (A\boldsymbol{z} + \delta^{-1} \cdot A\boldsymbol{z'}) \circ \boldsymbol{\gamma}^{m} - \boldsymbol{\beta}^{m}, B\boldsymbol{z} - \boldsymbol{\alpha}^{m} + \delta^{-1} \cdot B\boldsymbol{z'} \rangle .$$

$$(4.1)$$

Focusing on each of the two inner products on the right-hand side of Eq. (4.1), for the first one we obtain

$$\langle (\boldsymbol{x'}||\boldsymbol{y}) + \delta^{-1} \cdot (\boldsymbol{x}||\boldsymbol{y'}) + \delta^{2} \cdot \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle$$

$$= \langle (\delta^{-1}\boldsymbol{x} \mid | \boldsymbol{y}), \boldsymbol{c} \circ \boldsymbol{\delta} \rangle + \langle (\boldsymbol{x'} \mid | \delta^{-1}\boldsymbol{y'}), \boldsymbol{c} \circ \boldsymbol{\delta} \rangle + \delta^{2} \cdot \langle \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle$$

$$= \langle \boldsymbol{\delta}^{-1} \circ \boldsymbol{z}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle + \delta \cdot \langle \boldsymbol{x'}, \boldsymbol{c}_{[1:r]} \rangle + \delta^{-1} \cdot \langle \boldsymbol{y'}, \boldsymbol{c}_{[r+1:n]} \rangle + \delta^{2} \cdot \langle \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle$$

$$= \langle \boldsymbol{z}, \boldsymbol{c} \rangle + \delta^{-1} \cdot \langle \boldsymbol{y'}, \boldsymbol{c}_{[r+1:n]} \rangle + \delta^{2} \cdot \langle \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle , \qquad (4.2)$$

where Eq. (4.2) relies on the fact that $A_{[1:r]}x' = B_{[1:r]}x' = C_{[1:r]}x' = 0^m$ that combined with $c = \mu^m A + \beta^m B - \gamma^m C$ implies

$$\langle \boldsymbol{x'}, \boldsymbol{c}_{[1:r]} \rangle = \langle \boldsymbol{x'}, \boldsymbol{\mu}^m A_{[1:r]} + \boldsymbol{\beta}^m B_{[1:r]} - \boldsymbol{\gamma}^m C_{[1:r]} \rangle = 0$$
.

For the second inner-product on the right-hand side of Eq. (4.1), we obtain

$$\langle (Az + \delta^{-1} \cdot Az') \circ \boldsymbol{\gamma}^{m} - \boldsymbol{\beta}^{m}, Bz - \boldsymbol{\alpha}^{m} + \delta^{-1} \cdot Bz' \rangle$$

$$= \langle \boldsymbol{\alpha}^{m}, \boldsymbol{\beta}^{m} \rangle + \delta^{-1} \cdot \boldsymbol{\gamma}^{m} (Az \circ Bz' + Bz \circ Az') - \delta^{-1} \cdot (\boldsymbol{\alpha}^{m} \circ \boldsymbol{\gamma}^{m}) Az' - \delta^{-1} \boldsymbol{\beta}^{m} Bz'$$

$$+ \boldsymbol{\gamma}^{m} (Az \circ Bz) - (\boldsymbol{\alpha}^{m} \circ \boldsymbol{\gamma}^{m}) Az - \boldsymbol{\beta}^{m} Bz + \delta^{-2} \cdot \boldsymbol{\gamma}^{m} (Az' \circ Bz')$$

$$= \langle \boldsymbol{\alpha}^{m}, \boldsymbol{\beta}^{m} \rangle + \delta^{-1} \cdot (\boldsymbol{\gamma}^{m} Cz' - \boldsymbol{\mu}^{m} Az' - \boldsymbol{\beta}^{m} Bz') + \boldsymbol{\gamma}^{m} Cz - \boldsymbol{\mu}^{m} Az - \boldsymbol{\beta}^{m} Bz \quad (4.3)$$

$$= \langle \boldsymbol{\alpha}^{m}, \boldsymbol{\beta}^{m} \rangle - \delta^{-1} \cdot \langle \boldsymbol{z}', \boldsymbol{c} \rangle - \langle \boldsymbol{z}, \boldsymbol{c} \rangle$$

$$= \langle \boldsymbol{\alpha}^{m}, \boldsymbol{\beta}^{m} \rangle - \delta^{-1} \cdot \langle \boldsymbol{y}', \boldsymbol{c}_{[r+1:n]} \rangle - \langle \boldsymbol{z}, \boldsymbol{c} \rangle \quad (4.4)$$

where Eq. (4.3) follows from the facts that $A\boldsymbol{z} \circ B\boldsymbol{z} = C\boldsymbol{z}$, $A\boldsymbol{z'} \circ B\boldsymbol{z'} = 0^m$ and $A\boldsymbol{z} \circ B\boldsymbol{z'} + B\boldsymbol{z} \circ A\boldsymbol{z'} = C\boldsymbol{z'}$, and Eq. (4.4) follows from the fact that $A_{[1:r]}\boldsymbol{x'} = B_{[1:r]}\boldsymbol{x'} = C_{[1:r]}\boldsymbol{x'} = 0^m$ implies $\langle \boldsymbol{z'}, \boldsymbol{c} \rangle = \langle \boldsymbol{y'}, \boldsymbol{c}_{[r+1:n]} \rangle$. Combining Eq. (4.1), (4.2) and (4.4), we obtain

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \langle \boldsymbol{\alpha}^m, \boldsymbol{\beta}^m \rangle + \delta^2 \cdot \langle \boldsymbol{\alpha}^n, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle = \omega$$
.

Second, by the guarantee $T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) \mid\mid A\boldsymbol{z'}), \boldsymbol{G} \rangle + \langle (0^n \mid\mid B\boldsymbol{z'}), \boldsymbol{H} \rangle + \eta \cdot H$, and by definitions of the group elements S and P as computed in the argument system, we obtain

$$P = \delta^{-1} \cdot T + S + \langle (\delta^{2} \cdot \boldsymbol{\alpha}^{n} \mid | -\beta^{m}), \boldsymbol{G'} \rangle + \langle (\boldsymbol{c} \circ \boldsymbol{\delta} \mid | -\boldsymbol{\alpha}^{m}), \boldsymbol{H} \rangle$$

$$= \delta^{-1} \cdot \langle ((\boldsymbol{x} | | \boldsymbol{y'}) \mid | A\boldsymbol{z'}), \boldsymbol{G} \rangle + \delta^{-1} \cdot \langle (0^{n} \mid | B\boldsymbol{z'}), \boldsymbol{H} \rangle + \delta^{-1} \cdot \eta \cdot H$$

$$+ \langle ((\boldsymbol{x'} | | \boldsymbol{y}) \mid | A\boldsymbol{z}), \boldsymbol{G} \rangle + \langle (0^{n} \mid | B\boldsymbol{z}), \boldsymbol{H} \rangle + r \cdot H$$

$$+ \langle (\delta^{2} \cdot \boldsymbol{\alpha}^{n} \mid | -\beta^{m}), \boldsymbol{G'} \rangle + \langle (\boldsymbol{c} \circ \boldsymbol{\delta} \mid | -\boldsymbol{\alpha}^{m}), \boldsymbol{H} \rangle$$

$$= \langle ((\boldsymbol{x'} | | \boldsymbol{y}) + \delta^{-1} \cdot (\boldsymbol{x} | | \boldsymbol{y'}) + \delta^{2} \cdot \boldsymbol{\alpha}^{n} \mid | (A\boldsymbol{z} + \delta^{-1} \cdot A\boldsymbol{z'}) \circ \boldsymbol{\gamma}^{m} - \boldsymbol{\beta}^{m}), \boldsymbol{G'} \rangle$$

$$+ \langle (\boldsymbol{c} \circ \boldsymbol{\delta} \mid | B\boldsymbol{z} - \boldsymbol{\alpha}^{m} + \delta^{-1} \cdot B\boldsymbol{z'}), \boldsymbol{H} \rangle + (r + \delta^{-1} \cdot \eta) \cdot H$$

$$= \langle \boldsymbol{u}, \boldsymbol{G'} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \eta' \cdot H .$$

Overall, we showed that $\omega = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$ and $P = \langle \boldsymbol{u}, \boldsymbol{G'} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \eta' \cdot H$, and therefore $((\boldsymbol{G'}, \boldsymbol{H}, G, H, P, \omega), (\boldsymbol{u}, \boldsymbol{v}, \eta')) \in \mathcal{R}_{IP}$. Thus, the perfect completeness of the argument system Π_{IP} implies that the argument system Π_{R1CS^*} has perfect completeness.

4.2 Honest-Verifier Zero-Knowledge

We prove the following lemma by showing that the honest-verifier zero-knowledge of the argument system Π_{R1CS^*} is directly inherited from that of the inner-product argument system Π_{IP} .

Lemma 4.3. The argument system Π_{RICS^*} has perfect special honest-verifier zero-knowledge.

Proof. Let S be the simulator that is defined as follows on input $((G, H, G, H), (T, A, B, C), \rho)$:

- 1. Uniformly sample $S \leftarrow \mathbb{G}$.
- 2. Parse $\rho = (\alpha, \beta, \gamma, \delta, \rho_{\text{IP}})$, where $\alpha, \beta, \gamma, \delta \in \mathbb{Z}_q^*$ is the verifier's randomness for Step 2 of the argument system Π_{R1CS^*} , and ρ_{IP} is the verifier's randomness for Step 4 of the argument system Π_{R1CS^*} (i.e., ρ_{IP} is the verifier's randomness for the argument system Π_{IP}).

3. Compute

$$\mu = \alpha \cdot \gamma \in \mathbb{Z}_{q}$$

$$\boldsymbol{\delta} = (\delta, \dots, \delta, 1^{n-r}) \in \mathbb{Z}_{q}^{n}$$

$$\boldsymbol{G'} = (G_{1}, \dots, G_{n}, \gamma^{-1} \cdot G_{n+1}, \dots, \gamma^{-m} \cdot G_{n+m}) \in \mathbb{G}^{n+m}$$

$$\boldsymbol{c} = \boldsymbol{\mu}^{m} A + \boldsymbol{\beta}^{m} B - \boldsymbol{\gamma}^{m} C \in \mathbb{Z}_{q}^{n}$$

$$\boldsymbol{\omega} = \langle \boldsymbol{\alpha}^{m}, \boldsymbol{\beta}^{m} \rangle + \delta^{2} \cdot \langle \boldsymbol{\alpha}^{n}, \boldsymbol{c} \circ \boldsymbol{\delta} \rangle \in \mathbb{Z}_{q}$$

$$P = \delta^{-1} \cdot T + S + \langle (\delta^{2} \cdot \boldsymbol{\alpha}^{n} \mid | -\boldsymbol{\beta}^{m}), \boldsymbol{G'} \rangle + \langle (\boldsymbol{c} \circ \boldsymbol{\delta} \mid | -\boldsymbol{\alpha}^{m}), \boldsymbol{H} \rangle \in \mathbb{G}$$

- 4. Invoke the zero-knowledge simulator of the argument system $\Pi_{\rm IP}$ on input $((\mathbf{G'}, \mathbf{H'}, G, H, P, \omega), \rho_{\rm IP})$ for obtaining a transcript $\mathsf{tr}_{\rm IP}$.
- 5. Output $(S, \alpha, \beta, \gamma, \delta, \mathsf{tr}_{\mathsf{IP}})$.

Let A_1 and A_2 be any two algorithms as in Definition 2.3, fix any common-reference string $\sigma = (G, H, G, H)$, and fix any triplet $((T, A, B, C), (\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta), \rho)$ produced by $A_1(\sigma)$ such that $((T, A, B, C), (\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta)) \in \mathcal{R}_{R1CS^*}$. We need to show that, conditioned on any such fixed values, the distribution of the transcript produced by the simulator S is identical to the distribution of an honestly-generated transcript (thus, A_2 will have no advantage in distinguishing the two cases – as required by Definition 2.3).

First, in both cases, the group element S is uniformly distributed: For the transcript produced by the simulator S this follows directly from the fact that S uniformly samples $S \leftarrow \mathbb{G}$, and for an honestly-generated transcript this follows from the fact that the honest prover uniformly samples $r \leftarrow \mathbb{Z}_q$ and computes

$$S = \langle ((\boldsymbol{x'}||\boldsymbol{y}) \mid |A\boldsymbol{z}), \boldsymbol{G} \rangle + \langle (0^n \mid |B\boldsymbol{z}), \boldsymbol{H} \rangle + r \cdot H.$$

Second, in both cases, the values α , β , γ , and δ are uniquely determined by the verifier's randomness $\rho = (\alpha, \beta, \gamma, \delta, \rho_{\text{IP}})$. Finally, in both cases the inner-product instance $(\mathbf{G'}, \mathbf{H}, G, H, P, \omega)$ is computed as the same deterministic function of the given generators $(\mathbf{G}, \mathbf{H}, G, H)$ and of $(S, \alpha, \beta, \gamma, \delta)$. Given that $((T, A, B, C), (\mathbf{x}, \mathbf{x'}, \mathbf{y}, \mathbf{y'}, \eta)) \in \mathcal{R}_{\text{R1CS}^*}$, the perfect completeness of the argument system Π_{R1CS^*} guarantees that there exists a witness $(\mathbf{u}, \mathbf{v}, \eta')$ (which would be computed by the prover in an honest execution as instructed) such that $((\mathbf{G'}, \mathbf{H}, G, H, P, \omega), (\mathbf{u}, \mathbf{v}, \eta')) \in \mathcal{R}_{\text{IP}}$. Thus, the perfect special honest-verifier zero-knowledge of the argument system Π_{IP} guarantees that the transcript tr_{IP} produced by its corresponding simulator is distributed identically to an honestly-generated transcript when conditioned on all previously-fixed values.

4.3 Witness-Extended Emulation

We prove that the argument system Π_{R1CS^*} provides computational witness-extended emulation based on the sufficient condition established by the general forking lemma of Bootle et al. (as discussed in Section 2.1 above). That is, we present a probabilistic polynomial-time algorithm that when provided with a transcript tree (of a suitable polynomial size) extracts either a valid witness or a non-trivial discrete-logarithm relation for (G, H, G, H).

Lemma 4.4. There exists a probabilistic polynomial-time algorithm Ext that, on input any $(G, H, G, H) \in \mathbb{G}^{2(n+m)+2}$ and any \mathcal{R}_{R1CS^*} instance (T, A, B, C) together with any corresponding $(n + 1, m + 1, m + 1, 5, 2, 4, \dots, 4, 5)$ -transcript tree of depth $\log_2(n + m) + 6$ for the argument system Π_{R1CS^*} , produces either a witness $(\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta)$ such that $((T, A, B, C), (\boldsymbol{x}, \boldsymbol{x'}, \boldsymbol{y}, \boldsymbol{y'}, \eta)) \in \mathcal{R}_{R1CS^*}$ or a non-trivial discrete-logarithm relation for (G, H, G, H).

Proof. Let $m, r, n \in \mathbb{N}$ be such that $m \geq 1$, $1 \leq r \leq n$ and $m + n = 2^t$, let \mathbb{G} be a cyclic group of prime order q, and let $G, H \in \mathbb{G}^{n+m}$ and $G, H \in \mathbb{G}$ be 2(n+m)+2 generators. Then, any $(n+1, m+1, m+1, 5, 2, 4, \ldots, 4, 5)$ -transcript tree of depth $\log_2(n+m)+6$ for an \mathcal{R}_{R1CS^*} instance (T, A, B, C) has the following form:

- The root of the tree is a group element $S \in \mathbb{G}$, and the first level consists of n+1 nodes corresponding to distinct values $\{\alpha_i\}_{i\in[n+1]}$.
- For each first-level node α_i , the second level consists of m+1 children corresponding to distinct values $\{\beta_{i,j}\}_{j\in[m+1]}$.
- For each second-level node $\beta_{i,j}$, the third level consists of m+1 children corresponding to distinct values $\{\gamma_{i,j,k}\}_{k\in[m+1]}$.
- For each third-level node $\gamma_{i,j,k}$, the forth level consists of 5 children corresponding to distinct values $\{\delta_{i,j,k,\ell}\}_{\ell\in[5]}$.
- Finally, each forth level node $\delta_{i,j,k,\ell}$ serves as the root of a $(2,4,\ldots,4,5)$ -transcript sub-tree of depth $\log_2(n+m)+2$ for the inner product argument with a corresponding instance $(G'_{i,j,k}, H, G, H, P_{i,j,k,\ell}, \omega_{i,j,k,\ell})$.

Our extractor Ext invokes the probabilistic polynomial-time extractor of the inner-product argument system (recall Lemma 2.7) on each of the $(2,4,\ldots,4,5)$ -transcript sub-trees (note that the number of such sub-trees is polynomial). For each such sub-tree it obtains either a witness $(\boldsymbol{u_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}) \in \mathcal{R}_{\text{IP}}$, or a non-trivial discrete-logarithm relation for $(\boldsymbol{G'_{i,j,k}}, \boldsymbol{H}, \boldsymbol{G}, \boldsymbol{H})$. Any such relation yields a corresponding relation for $(\boldsymbol{G}, \boldsymbol{H}, \boldsymbol{G}, \boldsymbol{H})$, and therefore for the remainder of the proof we assume that the extractor obtains a witness $(\boldsymbol{u_{i,j,k,\ell}}, \boldsymbol{v_{i,j,k,\ell}}, \eta'_{i,j,k,\ell})$ for each such sub-tree.

For every $i \in [n+1]$, $j, k \in [m+1]$ and $\ell \neq \ell' \in [5]$, note that for the \mathcal{R}_{IP} instances corresponding to the two paths $(\alpha_i, \beta_{i,j}, \gamma_{i,j,k}, \delta_{i,j,k,\ell})$ and $(\alpha_i, \beta_{i,j}, \gamma_{i,j,k}, \delta_{i,j,k,\ell'})$ it holds that

$$S + \delta_{i,j,k,\ell}^{-1} \cdot T = P_{i,j,k,\ell} - \langle (\delta_{i,j,k,\ell}^2 \cdot \boldsymbol{\alpha}_i^n \mid | - \boldsymbol{\beta}_{i,j}^m), \boldsymbol{G}_{i,j,k}' \rangle - \langle (\boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell} \mid | - \boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle$$

$$= \langle \boldsymbol{u}_{i,j,k,\ell}, \boldsymbol{G}_{i,j,k}' \rangle + \langle \boldsymbol{v}_{i,j,k,\ell}, \boldsymbol{H} \rangle + \eta'_{i,j,k,\ell} \cdot \boldsymbol{H}$$

$$- \langle (\delta_{i,j,k,\ell}^2 \cdot \boldsymbol{\alpha}_i^n \mid | - \boldsymbol{\beta}_{i,j}^m), \boldsymbol{G}_{i,j,k}' \rangle - \langle (\boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell} \mid | - \boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle$$

$$S + \delta_{i,j,k,\ell'}^{-1} \cdot T = P_{i,j,k,\ell'} - \langle (\delta_{i,j,k,\ell'}^2 \cdot \boldsymbol{\alpha}_i^n \mid | - \boldsymbol{\beta}_{i,j}^m), \boldsymbol{G}_{i,j,k}' \rangle - \langle (\boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell'} \mid | - \boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle$$

$$= \langle \boldsymbol{u}_{i,j,k,\ell'}, \boldsymbol{G}_{i,j,k}' \rangle + \langle \boldsymbol{v}_{i,j,k,\ell'}, \boldsymbol{H} \rangle + \eta'_{i,j,k,\ell'} \cdot \boldsymbol{H}$$

$$- \langle (\delta_{i,j,k,\ell'}^2 \cdot \boldsymbol{\alpha}_i^n \mid | - \boldsymbol{\beta}_{i,j}^m), \boldsymbol{G}_{i,j,k}' \rangle - \langle (\boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell'} \mid | - \boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle ,$$

As $\delta_{i,j,k,\ell} \neq \delta_{i,j,k,\ell'}$, this provides two linearly-independent equations in the unknowns S and T, and enables to efficiently compute vectors

$$egin{aligned} s_{i,j,k,\ell,\ell',1}, s_{i,j,k,\ell,\ell',1}', t_{i,j,k,\ell,\ell',1}, t_{i,j,k,\ell,\ell',1}' &\in \mathbb{Z}_q^r \ s_{i,j,k,\ell,\ell',2}, s_{i,j,k,\ell,\ell',2}', t_{i,j,k,\ell,\ell',2}, t_{i,j,k,\ell,\ell',2}' &\in \mathbb{Z}_q^{n-r} \ s_{i,j,k,\ell,\ell',3}, s_{i,j,j',3}', t_{i,j,k,\ell,\ell',3}, t_{i,j,k,\ell,\ell',3}', t_{i,j,k,\ell,\ell',3}' &\in \mathbb{Z}_q^m \ s_{i,i,k,\ell,\ell'}', t_{i,j,k,\ell,\ell'}'' &\in \mathbb{Z}_q \end{aligned}$$

such that

$$S = \langle s_{i,j,k,\ell,\ell',1} || s_{i,j,k,\ell,\ell',2} || s_{i,j,k,\ell,\ell',3}, G \rangle$$

$$+ \langle s'_{i,j,k,\ell,\ell',1} || s'_{i,j,k,\ell,\ell',2} || s'_{i,j,k,\ell,\ell',3}, H \rangle + s''_{i,j,k,\ell,\ell'} \cdot H$$

$$T = \langle t_{i,j,k,\ell,\ell',1} || t_{i,j,k,\ell,\ell',2} || t_{i,j,k,\ell,\ell',3}, G \rangle$$

$$+ \langle t'_{i,j,k,\ell,\ell',1} || t'_{i,j,k,\ell,\ell',2} || t'_{i,j,k,\ell,\ell',3}, H \rangle + t''_{i,j,k,\ell,\ell'} \cdot H$$

If these vectors are not identical for all (i, j, k, ℓ, ℓ') , then we obtain a non-trivial discrete-logarithm relation for (G, H, H) (and thus a non-trivial discrete-logarithm relation for (G, H, G, H)). Therefore, for the remainder of the proof we assume that these vectors are identical for all (i, j, k, ℓ, ℓ') , and denote them by $s_1, s'_1, t_1, t'_1 \in \mathbb{Z}_q^r$, $s_2, s'_2, t_2, t'_2 \in \mathbb{Z}_q^{n-r}$, $s_3, s'_3, t_3, t'_3 \in \mathbb{Z}_q^m$ and $s'', t'' \in \mathbb{Z}_q$. Equipped with this notation, we have

$$S = \langle s_1 || s_2 || s_3, G \rangle + \langle s_1' || s_2' || s_3', H \rangle + s'' \cdot H$$

$$T = \langle t_1 || t_2 || t_3, G \rangle + \langle t_1' || t_2' || t_3', H \rangle + t'' \cdot H$$

Next, for every $i \in [n+1]$, $j,k \in [m+1]$ and $\ell \in [5]$, the extracted witness $(u_{i,j,k,\ell}, v_{i,j,k,\ell}, \eta'_{i,j,k,\ell})$ satisfies

$$\begin{split} \langle \boldsymbol{u}_{i,j,k,\ell}, \boldsymbol{G}'_{i,j,k} \rangle + \langle \boldsymbol{v}_{i,j,k,\ell}, \boldsymbol{H} \rangle + \eta'_{i,j,k,\ell} \cdot \boldsymbol{H} \\ &= P_{i,j,k,\ell} \\ &= S + \delta_{i,j,k,\ell}^{-1} \cdot T + \langle (\delta_{i,j,k,\ell}^2 \cdot \boldsymbol{\alpha}_i^n \mid\mid -\beta_{i,j}^m), \boldsymbol{G}'_{i,j,k} \rangle + \langle (\boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell} \mid\mid -\boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle \\ &= \langle \boldsymbol{s}_1 \mid\mid \boldsymbol{s}_2 \mid\mid \boldsymbol{s}_3, \boldsymbol{G} \rangle + \langle \boldsymbol{s}'_1 \mid\mid \boldsymbol{s}'_2 \mid\mid \boldsymbol{s}'_3 \mid\mid \boldsymbol{H} \rangle + \boldsymbol{s}'' \cdot \boldsymbol{H} \\ &+ \delta_{i,j,k,\ell}^{-1} \cdot \left(\langle \boldsymbol{t}_1 \mid\mid \boldsymbol{t}_2 \mid\mid \boldsymbol{t}_3, \boldsymbol{G} \rangle + \langle \boldsymbol{t}'_1 \mid\mid \boldsymbol{t}'_2 \mid\mid \boldsymbol{t}'_3, \boldsymbol{H} \rangle + \boldsymbol{t}'' \cdot \boldsymbol{H} \right) \\ &+ \langle (\delta_{i,j,k,\ell}^2 \cdot \boldsymbol{\alpha}_i^n \mid\mid -\beta_{i,j}^m), \boldsymbol{G}'_{i,j,k} \rangle + \langle (\boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell} \mid\mid -\boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle \\ &= \langle (((\boldsymbol{s}_1 \mid\mid \boldsymbol{s}_2) + \delta_{i,j,k,\ell}^{-1} \cdot (\boldsymbol{t}_1 \mid\mid \boldsymbol{t}_2) + \delta_{i,j,k,\ell}^2 \cdot \boldsymbol{\alpha}_i^n) \mid\mid (\boldsymbol{s}_3 + \delta_{i,j,k,\ell}^{-1} \cdot \boldsymbol{t}_3) \circ \boldsymbol{\gamma}_{i,j,k}^m - \beta_{i,j}^m), \boldsymbol{G}'_{i,j,k} \rangle \\ &+ \langle (((\boldsymbol{s}'_1 \mid\mid \boldsymbol{s}'_2) + \delta_{i,j,k,\ell}^{-1} \cdot (\boldsymbol{t}'_1 \mid\mid \boldsymbol{t}'_2) + \boldsymbol{c}_{i,j,k} \circ \delta_{i,j,k,\ell}) \mid\mid \boldsymbol{s}'_3 + \delta_{i,j,k,\ell}^{-1} \cdot \boldsymbol{t}'_3 - \boldsymbol{\alpha}_i^m), \boldsymbol{H} \rangle \\ &+ (\boldsymbol{s}'' + \delta_{i,j,k,\ell}^{-1} \cdot \boldsymbol{t}'') \cdot \boldsymbol{H} \; . \end{split}$$

Therefore,

$$\begin{aligned} & \boldsymbol{u_{i,j,k,\ell}} = (((\boldsymbol{s_1}||\boldsymbol{s_2}) + \delta_{i,j,k,\ell}^{-1} \cdot (\boldsymbol{t_1}||\boldsymbol{t_2}) + \delta_{i,j,k,\ell}^2 \cdot \boldsymbol{\alpha_i^n}) \mid\mid (\boldsymbol{s_3} + \delta_{i,j,k,\ell}^{-1} \cdot \boldsymbol{t_3}) \circ \boldsymbol{\gamma_{i,j,k}^m} - \boldsymbol{\beta_{i,j}^m}) \\ & \boldsymbol{v_{i,j,k,\ell}} = (((\boldsymbol{s_1'}||\boldsymbol{s_2'}) + \delta_{i,j,k,\ell}^{-1} \cdot (\boldsymbol{t_1'}||\boldsymbol{t_2'}) + \boldsymbol{c_{i,j,k}} \circ \boldsymbol{\delta_{i,j,k,\ell}}) \mid\mid \boldsymbol{s_3'} + \delta_{i,j,k,\ell}^{-1} \cdot \boldsymbol{t_3'} - \boldsymbol{\alpha_i^m}) \\ & \boldsymbol{\eta_{i,j,k,\ell}'} = \boldsymbol{s_1''} + \delta_{i,j,k,\ell}^{-1} \cdot \boldsymbol{t_1''}, \end{aligned}$$

since otherwise we again obtain a non-trivial discrete-logarithm relation for $(G'_{i,j,k}, H, H)$ (and thus

a non-trivial discrete-logarithm relation for (G, H, G, H)). Then, on the one hand

$$\begin{split} \omega_{i,j,k,\ell} &= \langle \boldsymbol{u}_{i,j,k,\ell}, \boldsymbol{v}_{i,j,k,\ell} \rangle \\ &= \left(\langle (\boldsymbol{t}_1 || \boldsymbol{t}_2), (\boldsymbol{t}_1' || \boldsymbol{t}_2') \rangle + \langle \boldsymbol{t}_3 \circ \boldsymbol{\gamma}_{i,j,k}^m, \boldsymbol{t}_3' \rangle \right) \cdot \delta_{i,j,k,\ell}^{-2} \\ &+ \left(\langle (\boldsymbol{s}_1 || \boldsymbol{s}_2), (\boldsymbol{t}_1' || \boldsymbol{t}_2') \rangle + \langle (\boldsymbol{s}_1' || \boldsymbol{s}_2'), (\boldsymbol{t}_1 || \boldsymbol{t}_2) \rangle + \langle \boldsymbol{t}_2, \boldsymbol{c}_{i,j,k,[r+1:n]} \rangle \right. \\ &+ \langle \boldsymbol{s}_3' - \boldsymbol{\alpha}_i^m, \boldsymbol{t}_3 \circ \boldsymbol{\gamma}_{i,j,k}^m \rangle + \langle \boldsymbol{s}_3 \circ \boldsymbol{\gamma}_{i,j,k}^m - \boldsymbol{\beta}_{i,j}^m, \boldsymbol{t}_3' \rangle \right) \cdot \delta_{i,j,k,\ell}^{-1} \\ &+ \left(\langle \boldsymbol{s}_1, \boldsymbol{s}_1' \rangle + \langle \boldsymbol{s}_2, (\boldsymbol{s}_2' + \boldsymbol{c}_{i,j,k,[r+1:n]}) \rangle + \langle \boldsymbol{t}_1, \boldsymbol{c}_{i,j,k,[1:r]} \rangle \right. \\ &+ \langle \boldsymbol{s}_3 \circ \boldsymbol{\gamma}_{i,j,k}^m, \boldsymbol{s}_3' - \boldsymbol{\alpha}_i^m \rangle - \langle \boldsymbol{\beta}_{i,j}^m, \boldsymbol{s}_3' \rangle + \langle \boldsymbol{\alpha}_i^m, \boldsymbol{\beta}_{i,j}^m \rangle \right) \\ &+ \left(\langle \boldsymbol{s}_1, \boldsymbol{c}_{i,j,k,[1:r]} \rangle + \langle \boldsymbol{\alpha}_i^n, (\boldsymbol{t}_1' || \boldsymbol{t}_2') \rangle \right) \cdot \delta_{i,j,k,\ell} \\ &+ \left(\langle \boldsymbol{\alpha}_i^n, (\boldsymbol{s}_1' || \boldsymbol{s}_2') \rangle + \langle \boldsymbol{\alpha}_i^r \cdot \boldsymbol{\alpha}_i^{n-r}, \boldsymbol{c}_{i,j,k,[r+1:n]} \rangle \right) \cdot \delta_{i,j,k,\ell}^2 \\ &+ \langle \boldsymbol{\alpha}_i^r, \boldsymbol{c}_{i,j,k,[1:r]} \rangle \cdot \delta_{i,j,k,\ell}^3, \end{split}$$

whereas on the other hand

$$\omega_{i,j,k,\ell} = \langle \boldsymbol{\alpha}_{i}^{m}, \boldsymbol{\beta}_{i,j}^{m} \rangle + \delta_{i,j,k,\ell}^{2} \cdot \langle \boldsymbol{\alpha}_{i}^{n}, \boldsymbol{c}_{i,j,k} \circ \boldsymbol{\delta}_{i,j,k,\ell} \rangle$$

$$= \langle \boldsymbol{\alpha}_{i}^{m}, \boldsymbol{\beta}_{i,j}^{m} \rangle + \langle \boldsymbol{\alpha}_{i}^{r} \cdot \boldsymbol{\alpha}_{i}^{n-r}, \boldsymbol{c}_{i,j,k,[r+1:n]} \rangle \cdot \delta_{i,j,k,\ell}^{2} + \langle \boldsymbol{\alpha}_{i}^{r}, \boldsymbol{c}_{i,j,k,[1:r]} \rangle \cdot \delta_{i,j,k,\ell}^{3}$$

which together imply

$$0 = \left(\langle (t_1||t_2), (t'_1||t'_2) \rangle + \langle t_3 \circ \gamma^m_{i,j,k}, t'_3 \rangle \right) \cdot \delta^{-2}_{i,j,k,\ell}$$

$$+ \left(\langle (s_1||s_2), (t'_1||t'_2) \rangle + \langle (s'_1||s'_2), (t_1||t_2) \rangle + \langle t_2, c_{i,j,k,[r+1:n]} \rangle \right)$$

$$+ \langle s'_3 - \alpha^m_i, t_3 \circ \gamma^m_{i,j,k} \rangle + \langle s_3 \circ \gamma^m_{i,j,k} - \beta^m_{i,j}, t'_3 \rangle \right) \cdot \delta^{-1}_{i,j,k,\ell}$$

$$+ \left(\langle s_1, s'_1 \rangle + \langle s_2, (s'_2 + c_{i,j,k,[r+1:n]}) \rangle + \langle t_1, c_{i,j,k,[1:r]} \rangle + \langle s_3 \circ \gamma^m_{i,j,k}, s'_3 - \alpha^m_i \rangle - \langle \beta^m_{i,j}, s'_3 \rangle \right)$$

$$+ \left(\langle s_1, c_{i,j,k,[1:r]} \rangle + \langle \alpha^n_i, (t'_1||t'_2) \rangle \right) \cdot \delta_{i,j,k,\ell}$$

$$+ \langle \alpha^n_i, (s'_1||s'_2) \rangle \cdot \delta^2_{i,j,k,\ell} .$$

For every $i \in [n+1]$ and $j,k \in [m+1]$, the right-hand side of above equation (when multiplied by $\delta_{i,j,k,\ell}^2$) is a polynomial of degree 4 in the variable δ . Since the above holds for 5 distinct values $\{\delta_{i,j,k,\ell}\}_{\ell \in [5]}$, it is the zero polynomial, and therefore its 5 coefficients are all zeros. That is,

$$0 = \langle (t_1||t_2), (t_1'||t_2')\rangle + \langle t_3 \circ \gamma_{i,j,k}^m, t_3'\rangle$$

$$\tag{4.5}$$

$$0 = \langle (s_1||s_2), (t'_1||t'_2) \rangle + \langle (s'_1||s'_2), (t_1||t_2) \rangle + \langle t_2, c_{i,j,k,[r+1:n]} \rangle + \langle s'_3 - \alpha_i^m, t_3 \circ \gamma_{i,j,k}^m \rangle + \langle s_3 \circ \gamma_{i,j,k}^m - \beta_{i,j}^m, t'_3 \rangle$$

$$(4.6)$$

$$0 = \langle s_1, s_1' \rangle + \langle s_2, (s_2' + c_{i,j,k,[r+1:n]}) \rangle + \langle t_1, c_{i,j,k,[1:r]} \rangle + \langle s_3 \circ \gamma_{i,j,k}^m, s_3' - \alpha_i^m \rangle - \langle \beta_{i,j}^m, s_3' \rangle$$

$$(4.7)$$

$$0 = \langle s_1, c_{i,j,k,[1:r]} \rangle + \langle \alpha_i^n, (t_1'||t_2') \rangle$$

$$(4.8)$$

$$0 = \langle \boldsymbol{\alpha_i^n}, (\boldsymbol{s_1'} || \boldsymbol{s_2'}) \rangle . \tag{4.9}$$

Similarly, since Eq. (4.9) holds for n+1 distinct values of α_i , then $s'_1 = 0^r$ and $s'_2 = 0^{n-r}$. Eq. (4.7) now becomes

$$0 = \langle (t_1||s_2), c_{i,j,k} \rangle + \langle s_3 \circ \gamma_{i,j,k}^m, s_3' - \alpha_i^m \rangle - \langle \beta_{i,j}^m, s_3' \rangle$$

and recalling that $c_{i,j,k} = \mu_{i,i,k}^m A + \beta_{i,j}^m B - \gamma_{i,i,k}^m C$ and $\mu_{i,j,k} = \alpha_i \cdot \gamma_{i,j,k}$ we obtain

$$0 = \langle (t_1||s_2), \mu_{i,j,k}^m A + \beta_{i,j}^m B - \gamma_{i,j,k}^m C \rangle + \langle s_3 \circ \gamma_{i,j,k}^m, s_3' - \alpha_i^m \rangle - \langle \beta_{i,j}^m, s_3' \rangle$$

= $\langle A \cdot (t_1||s_2) - s_3, \alpha_i^m \circ \gamma_{i,j,k}^m \rangle + \langle B \cdot (t_1||s_2) - s_3', \beta_{i,j}^m \rangle - \langle C \cdot (t_1||s_2) - s_3' \circ s_3, \gamma_{i,j,k}^m \rangle$.

Applying Claim C.2 to the above, we obtain

$$A \cdot (t_1||s_2) = s_3$$

 $B \cdot (t_1||s_2) = s_3'$
 $C \cdot (t_1||s_2) = s_3' \circ s_3$

Letting $x = t_1 \in \mathbb{Z}_q^r$, $y = s_2 \in \mathbb{Z}_q^{n-r}$ and $z = (x||y) \in \mathbb{Z}_q^n$, yields $(Az) \circ (Bz) = Cz$ as required. Focusing now on the group elements S and T, so far we have established that they are of the following form:

$$S = \langle ((\mathbf{s_1}||\mathbf{y}) || A\mathbf{z}), \mathbf{G} \rangle + \langle (0^n || B\mathbf{z}), \mathbf{H} \rangle + s'' \cdot H$$

$$T = \langle ((\mathbf{z}||\mathbf{t_2}) || \mathbf{t_3}), \mathbf{G} \rangle + \langle ((\mathbf{t_1'}||\mathbf{t_2'}) || \mathbf{t_3'}), \mathbf{H} \rangle + t'' \cdot H.$$

Examining the term $\langle s_1, c_{i,j,k,[1:r]} \rangle$ that appears in Eq. (4.8), given that $c_{i,j,k} = \mu_{i,j,k}^m A + \beta_{i,j}^m B - \gamma_{i,j,k}^m C$, then

$$c_{i,j,k,[1:r]} = \mu_{i,j,k}^m A_{[1:r]} + \beta_{i,j}^m B_{[1:r]} - \gamma_{i,j,k}^m C_{[1:r]}$$

where $A_{[1:r]}$, $B_{[1:r]}$ and $C_{[1:r]} \in \mathbb{Z}_q^{m \times r}$ denote the leftmost r columns of the matrices A, B and C, respectively. Eq. (4.8) is thus equivalent to

$$0 = \langle A_{[1:r]} s_1, \alpha_i^m \circ \gamma_{i,i,k}^m \rangle + \langle B_{[1:r]} s_1, \beta_{i,i}^m \rangle - \langle C_{[1:r]} s_1, \gamma_{i,i,k}^m \rangle + \langle \alpha_i^n, (t_1' | | t_2') \rangle.$$

Applying Claim C.2 once again, we obtain $A_{[1:r]}\mathbf{s_1} = B_{[1:r]}\mathbf{s_1} = C_{[1:r]}\mathbf{s_1} = 0^m$, $\mathbf{t'_1} = 0^r$ and $\mathbf{t'_2} = 0^{n-r}$. Letting $\mathbf{x'} = \mathbf{s_1} \in \mathbb{Z}_q^r$ we thus have $A_{[1:r]}\mathbf{x'} = B_{[1:r]}\mathbf{x'} = C_{[1:r]}\mathbf{x'} = 0^m$ as required.

Similarly, examining the term $\langle t_2, c_{i,j,k,\lceil r+1:n\rceil} \rangle$ that appears in Eq. (4.6) it holds that

$$c_{i,j,k,[r+1:n]} = \mu_{i,j,k}^m A_{[r+1:n]} + \beta_{i,j}^m B_{[r+1:n]} - \gamma_{i,j,k}^m C_{[r+1:n]} ,$$

where $A_{[r+1:n]}$, $B_{[r+1:n]}$ and $C_{[r+1:n]} \in \mathbb{Z}_q^{m \times (n-r)}$ denote the rightmost n-r columns of the matrices A, B and C, respectively. Given that $s_1' = t_1' = 0^r$ and $s_2' = t_2' = 0^{n-r}$, it holds that $\langle (s_1||s_2), (t_1'||t_2') \rangle = 0$ and $\langle (s_1'||s_2'), (t_1||t_2) \rangle = 0$, and therefore from Eq. (4.6) we obtain

$$0 = \langle \boldsymbol{t_2}, \boldsymbol{c_{i,j,k,[r+1:n]}} \rangle + \langle \boldsymbol{s_3'} - \boldsymbol{\alpha_i^m}, \boldsymbol{t_3} \circ \boldsymbol{\gamma_{i,j,k}^m} \rangle + \langle \boldsymbol{s_3} \circ \boldsymbol{\gamma_{i,j,k}^m} - \boldsymbol{\beta_{i,j}^m}, \boldsymbol{t_3'} \rangle$$

$$= \langle A_{[r+1:n]} \boldsymbol{t_2}, \boldsymbol{\alpha_i^m} \circ \boldsymbol{\gamma_{i,j,k}^m} \rangle + \langle B_{[r+1:n]} \boldsymbol{t_2}, \boldsymbol{\beta_{i,j}^m} \rangle - \langle C_{[r+1:n]} \boldsymbol{t_2}, \boldsymbol{\gamma_{i,j,k}^m} \rangle$$

$$+ \langle \boldsymbol{s_3'} \circ \boldsymbol{t_3} + \boldsymbol{s_3} \circ \boldsymbol{t_3'}, \boldsymbol{\gamma_{i,j,k}^m} \rangle - \langle \boldsymbol{t_3}, \boldsymbol{\alpha_i^m} \circ \boldsymbol{\gamma_{i,j,k}^m} \rangle - \langle \boldsymbol{t_3'}, \boldsymbol{\beta_{i,j}^m} \rangle$$

$$= \langle A_{[r+1:n]} \boldsymbol{t_2} - \boldsymbol{t_3}, \boldsymbol{\alpha_i^m} \circ \boldsymbol{\gamma_{i,j,k}^m} \rangle + \langle B_{[r+1:n]} \boldsymbol{t_2} - \boldsymbol{t_3'}, \boldsymbol{\beta_{i,j}^m} \rangle - \langle C_{[r+1:n]} \boldsymbol{t_2} - \boldsymbol{s_3'} \circ \boldsymbol{t_3} - \boldsymbol{s_3} \circ \boldsymbol{t_3'}, \boldsymbol{\gamma_{i,j,k}^m} \rangle .$$

Applying Claim C.2 once again, it holds that

$$A_{[r+1:n]} t_2 = t_3$$

 $B_{[r+1:n]} t_2 = t_3'$
 $C_{[r+1:n]} t_2 = s_3' \circ t_3 + s_3 \circ t_3' = Bz \circ A_{[r+1:n]} t_2 + Az \circ B_{[r+1:n]} t_2$.

Letting $\mathbf{y'} = \mathbf{t_2} \in \mathbb{Z}_q^{n-r}$ and $\mathbf{z'} = (\mathbf{x'}||\mathbf{y'}) \in \mathbb{Z}_q^n$, we then obtain $C_{[r+1:n]}\mathbf{y'} = (B\mathbf{z} \circ A_{[r+1:n]}\mathbf{y'}) + (A\mathbf{z} \circ B_{[r+1:n]}\mathbf{y'})$, and thus $C\mathbf{z'} = (B\mathbf{z} \circ A\mathbf{z'}) + (A\mathbf{z} \circ B\mathbf{z'})$ as required.

Finally, examining Eq. (4.5), given that $\mathbf{t'_1} = 0^r$ and $\mathbf{t'_2} = 0^{n-r}$, then for m+1 distinct values of $\gamma_{i,j,k}$ it holds that $0 = \langle \mathbf{t_3} \circ \mathbf{t'_3}, \gamma_{i,j,k}^m \rangle$, and therefore $\mathbf{t_3} \circ \mathbf{t'_3} = 0^m$. That is, $(A_{[r+1:n]}\mathbf{y'}) \circ (B_{[r+1:n]}\mathbf{y'}) = 0^m$ and therefore $(A\mathbf{z'}) \circ (B\mathbf{z'}) = 0^m$. Letting $\eta = t''$, we obtain

$$T = \langle ((\boldsymbol{x}||\boldsymbol{y'}) \mid | A\boldsymbol{z'}), \boldsymbol{G} \rangle + \langle (0^n \mid | B\boldsymbol{z'}), \boldsymbol{H} \rangle + \eta \cdot H$$

as required.

Acknowledgments

We thank Arash Afshar, Dan Boneh, Yi-Hsiu Chen, Yehuda Lindell, Peihan Miao and Samuel Ranellucci for many useful discussions throughout this work.

References

- [AC20] T. Attema and R. Cramer. Compressed Σ-protocol theory and practical application to plug & play secure algorithmics. In D. Micciancio and T. Ristenpart, editors, Advances in Cryptology CRYPTO 2020, Part III, volume 12172 of Lecture Notes in Computer Science, pages 513–543, August 2020.
- [AFK23] T. Attema, S. Fehr, and M. Klooß. Fiat-Shamir transformation of multi-round interactive proofs (extended version). *Journal of Cryptology*, 36(4):36, October 2023.
- [BBB⁺18] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE Symposium on Security and Privacy*, pages 315–334. IEEE Computer Society Press, May 2018.
- [BCC⁺16] J. Bootle, A. Cerulli, P. Chaidos, J. Groth, and C. Petit. Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting. In M. Fischlin and J.-S. Coron, editors, *Advances in Cryptology EUROCRYPT 2016*, *Part II*, volume 9666 of *Lecture Notes in Computer Science*, pages 327–357, May 2016.
- [BCR+19] E. Ben-Sasson, A. Chiesa, M. Riabzev, N. Spooner, M. Virza, and N. P. Ward. Aurora: Transparent succinct arguments for R1CS. In Y. Ishai and V. Rijmen, editors, Advances in Cryptology – EUROCRYPT 2019, Part I, volume 11476 of Lecture Notes in Computer Science, pages 103–128, May 2019.
- [Bün23] B. Bünz. Improving the Privacy, Scalability, and Ecological Impact of Blockchains. PhD thesis, Stanford University, 2023. Available at https://cs.nyu.edu/~bb/papers/thesis.pdf.
- [CHJ⁺22] H. Chung, K. Han, C. Ju, M. Kim, and J. H. Seo. Bulletproofs+: Shorter proofs for a privacy-enhanced distributed ledger. *IEEE Access*, 10:42067–42082, 2022.
- [CHM⁺20] A. Chiesa, Y. Hu, M. Maller, P. Mishra, P. Vesely, and N. P. Ward. Marlin: Preprocessing zkSNARKs with universal and updatable SRS. In A. Canteaut and Y. Ishai, editors, Advances in Cryptology EUROCRYPT 2020, Part I, volume 12105 of Lecture Notes in Computer Science, pages 738–768, May 2020.

- [FS87] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. M. Odlyzko, editor, *Advances in Cryptology CRYPTO'86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194, August 1987.
- [GGP⁺13] R. Gennaro, C. Gentry, B. Parno, and M. Raykova. Quadratic span programs and succinct NIZKs without PCPs. In T. Johansson and P. Q. Nguyen, editors, *Advances in Cryptology EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 626–645, May 2013.
- [Gro16] J. Groth. On the size of pairing-based non-interactive arguments. In M. Fischlin and J.-S. Coron, editors, *Advances in Cryptology EUROCRYPT 2016, Part II*, volume 9666 of *Lecture Notes in Computer Science*, pages 305–326, May 2016.
- [Lin03] Y. Lindell. Parallel coin-tossing and constant-round secure two-party computation. *Journal of Cryptology*, 16(3):143–184, June 2003.
- [OB22] A. Ozdemir and D. Boneh. Experimenting with collaborative zk-SNARKs: Zero-knowledge proofs for distributed secrets. In K. R. B. Butler and K. Thomas, editors, USENIX Security 2022: 31st USENIX Security Symposium, pages 4291–4308. USENIX Association, August 2022.

A From mIP Arguments to IP Arguments

In this section we show that an argument system for the multiplicative inner-product relation \mathcal{R}_{mIP} can be used to construct an argument system for the inner-product relation \mathcal{R}_{IP} (see Section 2.2 for the definitions of these two relations). Given an argument system Π_{mIP} for the relation \mathcal{R}_{mIP} , consider the argument system Π_{IP} defined as follows:

The Argument System Π_{IP}

- Public parameters:
 - 1. Integer $d = 2^t \ge 1$.
 - 2. Cyclic group \mathbb{G} of prime order q.
- Inputs:
 - 1. \mathcal{P} : Instance $(G, H, G, H, P, \omega) \in \mathbb{G}^{2d+3} \times \mathbb{Z}_q$ and witness $(u, v, \alpha) \in \mathbb{Z}_q^{2d+1}$.
 - 2. \mathcal{V} : Instance $(\boldsymbol{G}, \boldsymbol{H}, G, H, P, \omega) \in \mathbb{G}^{2d+3} \times \mathbb{Z}_q$.
- Execution:
 - 1. The verifier \mathcal{V} samples $e \leftarrow \mathbb{Z}_q^*$, and sends e to the prover \mathcal{P} .
 - 2. Each party computes

$$G' = e \cdot G$$

$$P' = P + \omega \cdot G'$$

3. The parties invoke Π_{mIP} with the instance $(\boldsymbol{G}, \boldsymbol{H}, G', H, P')$, where the prover \mathcal{P} takes the role of the prover using the witness $(\boldsymbol{u}, \boldsymbol{v}, \alpha)$, and the verifier \mathcal{V} takes the role of the verifier and then outputs its output.

In what follows we prove the completeness, zero-knowledge and witness-extended emulation properties of the argument system Π_{IP} based on the corresponding properties of the underlying argument

system Π_{mIP} . Theorem 2.6 then follows by instantiating the underlying argument system Π_{mIP} with the argument system constructed by Chung et al. [CHJ⁺22] and described in Appendix B.

Claim A.1. Assuming that Π_{mIP} has perfect completeness, then Π_{IP} has perfect completeness.

Proof. Let \mathbb{G} be a cyclic group of prime order q, and let $((\boldsymbol{G}, \boldsymbol{H}, G, H, P, \omega), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{\text{IP}}$. We show that, for any $e \in \mathbb{Z}_q^*$ chosen by the verifier, it holds that $((\boldsymbol{G}, \boldsymbol{H}, G', H, P'), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{\text{mIP}}$ where $G' = e \cdot G$ and $P' = P + \omega \cdot G'$. Given that $((\boldsymbol{G}, \boldsymbol{H}, G, H, P, \omega), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{\text{IP}}$, then $P = \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \alpha \cdot H$ and $\omega = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$, and therefore

$$P' = P + \omega \cdot G'$$

= $\langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \alpha \cdot H + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot G'$

This implies that $((G, H, G', H, P'), (u, v, \alpha)) \in \mathcal{R}_{mIP}$ as required.

Claim A.2. Assuming that Π_{mIP} has perfect special honest-verifier zero-knowledge, then Π_{IP} has perfect special honest-verifier zero-knowledge.

Proof. Let S be the simulator that is defined as follows on input $((G, H, G, H, P, \omega), \rho)$:

- 1. Parse $\rho = (e, \rho_{\mathsf{mIP}})$, where $e \in \mathbb{Z}_q^*$ is the verifier's randomness for Step 1 of the argument system Π_{IP} , and ρ_{mIP} is the verifier's randomness for Step 3 of the argument system Π_{IP} (i.e., ρ_{mIP} is the verifier's randomness for the argument system Π_{mIP}).
- 2. Compute $G' = e \cdot G$ and $P' = P + \omega \cdot G'$.
- 3. Invoke the zero-knowledge simulator of the argument system Π_{mIP} on input $((\boldsymbol{G}, \boldsymbol{H}, G', H, P'), \rho_{\text{mIP}})$ for obtaining a transcript $\mathsf{tr}_{\mathsf{mIP}}$.
- 4. Output $(e, \mathsf{tr}_{\mathsf{mIP}})$.

Let A_1 and A_2 be any two algorithms as in Definition 2.3, and fix any common-reference string $\sigma = (\mathbb{G}, G, q)$ and any triplet $((G, H, G, H, P, \omega), (u, v, \alpha), \rho)$ produced by $A_1(\sigma)$ such that $((G, H, G, H, P, \omega), (u, v, \alpha)) \in \mathcal{R}_{IP}$. We need to show that, conditioned on any such fixed values, the distribution of the transcript produced by the simulator \mathcal{S} is identical to the distribution of an honestly-generated transcript (thus, A_2 will have no advantage in distinguishing the two cases – as required by Definition 2.3).

First, in both cases, the value e is uniquely determined by the verifier's randomness $\rho = (e, \rho_{\mathsf{mIP}})$. Second, in both cases, the $\mathcal{R}_{\mathsf{mIP}}$ instance (G, H, G', H, P') is computed as the same deterministic function of (G, H, G, H, P, ω) and e. Given that $((G, H, G, H, P, \omega), (u, v, \alpha)) \in \mathcal{R}_{\mathsf{IP}}$, the perfect completeness of the argument system Π_{IP} guarantees that $((G, H, G', H, P'), (u, v, \alpha)) \in \mathcal{R}_{\mathsf{mIP}}$. Thus, the perfect special honest-verifier zero-knowledge of the argument system Π_{mIP} guarantees that the transcript $\mathsf{tr}_{\mathsf{mIP}}$ produced by its corresponding simulator is distributed identically to an honestly-generated transcript when conditioned on all previously-fixed values.

Claim A.3. Let \mathbb{G} be a cyclic group of prime order q. Assume that Π_{mIP} is a $(2\mu + 1)$ -move public-coin argument system, and for each $i \in [\mu]$ let $n_i = n_i(\kappa) \geq 1$ such that $\Pi_{i=1}^{\mu} n_i$ is polynomial in the security parameter $\kappa \in \mathbb{N}$. Assume further that there exists a probabilistic polynomial-time algorithm Ext_{mIP} that when given any (n_1, \ldots, n_{μ}) -tree of accepting transcripts for an \mathcal{R}_{mIP} instance $(\mathbf{G}, \mathbf{H}, \mathbf{G}', \mathbf{H}, \mathbf{P}')$ always succeeds in extracting either a witness $(\mathbf{u}, \mathbf{v}, \alpha)$ such that $((\mathbf{G}, \mathbf{H}, \mathbf{G}', \mathbf{H}, \mathbf{P}'), (\mathbf{u}, \mathbf{v}, \alpha)) \in \mathcal{R}_{mIP}$ or a non-trivial discrete-logarithm relation for $(\mathbf{G}, \mathbf{H}, \mathbf{G}', \mathbf{H})$. Then, there exists

a probabilistic polynomial-time algorithm Ext_{IP} that when given any $(2, n_1, \ldots, n_\mu)$ -tree of accepting transcripts for an \mathcal{R}_{IP} instance $(\mathbf{G}, \mathbf{H}, G, H, P, \omega)$ always succeeds in extracting either a witness $(\mathbf{u}, \mathbf{v}, \alpha)$ such that $((\mathbf{G}, \mathbf{H}, G, H, P, \omega), (\mathbf{u}, \mathbf{v}, \alpha)) \in \mathcal{R}_{IP}$ or a non-trivial discrete-logarithm relation for $(\mathbf{G}, \mathbf{H}, G, H)$.

Proof. Any $(2, n_1, \ldots, n_{\mu})$ -transcript tree \mathcal{R}_{IP} for an \mathcal{R}_{IP} instance (G, H, G, H, P, ω) has the following form:

- The first level consists of 2 nodes corresponding to distinct values $e_1 \neq e_2$.
- Each second level node serves as the root of an (n_1, \ldots, n_μ) -transcript sub-tree for an \mathcal{R}_{mIP} instance (G, H, G'_i, H, P'_i) , where $G'_i = e_i \cdot G$ and $P' = P + \omega \cdot G'_i$ for $i \in \{1, 2\}$.

Consider the extractor $\mathsf{Ext}_{\mathsf{IP}}$ that invokes the given extractor $\mathsf{Ext}_{\mathsf{mIP}}$ on each of the two (n_1,\ldots,n_μ) -transcript sub-trees. By assumption, for each such sub-tree it obtains either a witness $(\boldsymbol{u_i},\boldsymbol{v_i},\alpha_i)$ such that $((\boldsymbol{G},\boldsymbol{H},G_i',H,P_i'),(\boldsymbol{u_i},v_i,\alpha_i))\in\mathcal{R}_{\mathsf{mIP}}$, or a non-trivial discrete-logarithm relation for $(\boldsymbol{G},\boldsymbol{H},G_i',H)$. Any such relation yields a corresponding relation for $(\boldsymbol{G},\boldsymbol{H},G,H)$, and therefore for the remainder of the proof we assume that the extractor obtains a witness $(\boldsymbol{u_i},\boldsymbol{v_i},\alpha_i)$ for each $i\in\{1,2\}$.

For each $i \in \{1, 2\}$ it thus holds that

$$P + \omega \cdot e_i \cdot G = P_i' = \langle \boldsymbol{u_i}, \boldsymbol{G} \rangle + \langle \boldsymbol{v_i}, \boldsymbol{H} \rangle + \langle \boldsymbol{u_i}, \boldsymbol{v_i} \rangle \cdot e_i \cdot G + \alpha_i \cdot H$$
(A.1)

and therefore

$$\omega \cdot (e_1 - e_2) \cdot G = \langle \boldsymbol{u_1} - \boldsymbol{u_2}, \boldsymbol{G} \rangle + \langle \boldsymbol{v_1} - \boldsymbol{v_2}, \boldsymbol{H} \rangle + (\langle \boldsymbol{u_1}, \boldsymbol{v_1} \rangle \cdot e_1 - \langle \boldsymbol{u_2}, \boldsymbol{v_2} \rangle \cdot e_2) \cdot G + (\alpha_1 - \alpha_2) \cdot H \quad (A.2)$$

If $u_1 \neq u_2$ or $v_1 \neq v_2$ or $\alpha_1 \neq \alpha_2$, then Eq. (A.2) yields a non-trivial discrete-logarithm relation for (G, H, G, H). Therefore, for the remainder of the proof we assume that $u_1 = u_2$, $v_1 = v_2$ and $\alpha_1 = \alpha_2$, and denote these values by u, v and α , respectively. Eq. (A.2) now simplifies to

$$\omega \cdot (e_1 - e_2) \cdot G = \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot (e_1 - e_2) \cdot G$$

which implies that $\omega = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$ since $e_1 \neq e_2$. Finally, from Eq. (A.1) we obtain

$$P = \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \alpha \cdot H$$
.

This implies that $((G, H, G, H, P, \omega), (u, v, \alpha)) \in \mathcal{R}_{IP}$ as required.

B The Chung et al. mIP Argument System

In this section we describe the argument system Π_{mIP} , constructed by Chung et al. [CHJ⁺22] for the relation \mathcal{R}_{mIP} (recall that we relied on their argument system in Appendix A as a building block for constructing an argument system for the relation \mathcal{R}_{IP}). Then, for completeness, we present the proof of security provided by Chung et al. when adapted to our notation and with some minor additional differences in the presentation.

The Argument System Π_{mIP}

- Public parameters:
 - 1. Integer $d = 2^t \ge 1$.
 - 2. Cyclic group \mathbb{G} of prime order q.
- Inputs:

- 1. \mathcal{P} : Instance $(G, H, G, H, P) \in \mathbb{G}^{2d+3}$ and witness $(u, v, \alpha) \in \mathbb{Z}_q^{2d+1}$.
- 2. V: Instance $(G, H, G, H, P) \in \mathbb{G}^{2d+3}$.
- Execution for d = 1:
 - 1. The prover \mathcal{P} samples $r, s, \delta, \eta \leftarrow \mathbb{Z}_q$, computes

$$A = r \cdot G + s \cdot H + (r \cdot v + s \cdot u) \cdot G + \delta \cdot H \in \mathbb{G}$$

$$B = (r \cdot s) \cdot G + \eta \cdot H \in \mathbb{G}$$

and sends A and B to the verifier \mathcal{V} .

- 2. The verifier \mathcal{V} samples $e \leftarrow \mathbb{Z}_q^*$, and sends e to the prover \mathcal{P} .
- 3. The prover \mathcal{P} computes

$$r' = r + \mathbf{u} \cdot e \in \mathbb{Z}_q$$

$$s' = s + \mathbf{v} \cdot e \in \mathbb{Z}_q$$

$$\delta' = \eta + \delta \cdot e + \alpha \cdot e^2 \in \mathbb{Z}_q$$

and sends r', s' and δ' to the verifier \mathcal{V} .

4. The verifier \mathcal{V} outputs accept if

$$e^2 \cdot P + e \cdot A + B = (r' \cdot e) \cdot G + (s' \cdot e) \cdot H + (r' \cdot s') \cdot G + \delta' \cdot H$$

and outputs reject otherwise.

- Execution for $d = 2^t > 1$:
 - 1. Let $\hat{d} = d/2$, $G = (G_1, G_2) \in \mathbb{G}^{\hat{d}} \times \mathbb{G}^{\hat{d}}$, $H = (H_1, H_2) \in \mathbb{G}^{\hat{d}} \times \mathbb{G}^{\hat{d}}$, $u = (u_1, u_2) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}$ and $v = (v_1, v_2) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}$.
 - 2. The prover \mathcal{P} samples $d_L, d_R \leftarrow \mathbb{Z}_q$, computes

$$c_{L} = \langle \boldsymbol{u}_{1}, \boldsymbol{v}_{2} \rangle \in \mathbb{Z}_{q}$$

$$c_{R} = \langle \boldsymbol{u}_{2}, \boldsymbol{v}_{1} \rangle \in \mathbb{Z}_{q}$$

$$L = \langle \boldsymbol{u}_{1}, \boldsymbol{G}_{2} \rangle + \langle \boldsymbol{v}_{2}, \boldsymbol{H}_{1} \rangle + c_{L} \cdot G + d_{L} \cdot H \in \mathbb{G}$$

$$R = \langle \boldsymbol{u}_{2}, \boldsymbol{G}_{1} \rangle + \langle \boldsymbol{v}_{1}, \boldsymbol{H}_{2} \rangle + c_{R} \cdot G + d_{R} \cdot H \in \mathbb{G}$$

and sends L and R to the verifier \mathcal{V} .

- 3. The verifier \mathcal{V} samples $e \leftarrow \mathbb{Z}_q^*$, and sends e to the prover \mathcal{P} .
- 4. Each party computes

$$\hat{\mathbf{G}} = e^{-1} \cdot \mathbf{G}_1 + e \cdot \mathbf{G}_2 \in \mathbb{G}^{\hat{d}}$$

$$\hat{\mathbf{H}} = e \cdot \mathbf{H}_1 + e^{-1} \cdot \mathbf{H}_2 \in \mathbb{G}^{\hat{d}}$$

$$\hat{P} = e^2 \cdot L + P + e^{-2} \cdot R \in \mathbb{G}$$

and the prover \mathcal{P} additionally computes

$$\hat{\boldsymbol{u}} = e \cdot \boldsymbol{u}_1 + e^{-1} \cdot \boldsymbol{u}_2 \in \mathbb{Z}_q^{\hat{d}}$$

$$\hat{\boldsymbol{v}} = e^{-1} \cdot \boldsymbol{v}_1 + e \cdot \boldsymbol{v}_2 \in \mathbb{Z}_q^{\hat{d}}$$

$$\hat{\alpha} = e^2 \cdot d_L + \alpha + e^{-2} \cdot d_R \in \mathbb{Z}_q$$

5. The parties invoke Π_{mIP} with the instance $(\hat{\boldsymbol{G}}, \hat{\boldsymbol{H}}, G, H, \hat{P})$, where the prover \mathcal{P} takes the role of the prover using the witness $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}, \hat{\alpha})$, and the verifier \mathcal{V} takes the role of the verifier and then outputs its output.

Chung et al. proved the following theorem, whose proof is provided via the proofs of Claims B.2, B.3 and B.4 below:

Theorem B.1 ([CHJ⁺22]). Let $t: \mathbb{N} \to \mathbb{N}$ be any function of the security parameter $\kappa \in \mathbb{N}$ such that $d = d(\kappa) = 2^{t(\kappa)}$ is polynomial. There exists an argument system Π_{IP} for the d-dimensional inner-product relation \mathcal{R}_{IP} that has perfect completeness, perfect special honest-verifier zero-knowledge, and statistical witness-extended emulation for extracting either a witness or a non-trivial discrete-logarithm relation. Furthermore, the argument system is public coin, and the prover communicates $2 \cdot \log_2 d + 2$ group elements and 3 field elements.

Claim B.2. The argument system Π_{mIP} has perfect completeness.

Proof. We prove via induction that the argument system Π_{mIP} has perfect completeness for instances of dimension $d=2^t$ for any integer $t\geq 0$. For t=0 (and thus d=1), let $(\boldsymbol{G},\boldsymbol{H},G,H,P)\in\mathbb{G}^{2d+3}$ and $(\boldsymbol{u},\boldsymbol{v},\alpha)\in\mathbb{Z}_q^{2d+1}$ be an instance and a witness, respectively, such that $((\boldsymbol{G},\boldsymbol{H},G,H,P),(\boldsymbol{u},\boldsymbol{v},\alpha))\in\mathcal{R}_{\text{mIP}}$. That is,

$$P = \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot G + \alpha \cdot H .$$

We show that, for any values $r, s, \delta, \eta \leftarrow \mathbb{Z}_q$ sampled by the prover and for any value $e \leftarrow \mathbb{Z}_q^*$ sampled by the verifier, it holds that

$$e^2 \cdot P + e \cdot A + B = (r' \cdot e) \cdot G + (s' \cdot e) \cdot H + (r' \cdot s') \cdot G + \delta' \cdot H$$

where

$$A = r \cdot G + s \cdot H + (r \cdot v + s \cdot u) \cdot G + \delta \cdot H \in \mathbb{G}$$

$$B = (r \cdot s) \cdot G + \eta \cdot H \in \mathbb{G}$$

$$r' = r + u \cdot e \in \mathbb{Z}_q$$

$$s' = s + v \cdot e \in \mathbb{Z}_q$$

$$\delta' = \eta + \delta \cdot e + \alpha \cdot e^2 \in \mathbb{Z}_q$$

are the values computed by the prover as instructed by the protocol. Indeed, it holds that

$$\begin{split} e^2 \cdot P + e \cdot A + B &= e^2 \cdot (\langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot G + \alpha \cdot H) \\ &+ e \cdot (r \cdot \boldsymbol{G} + s \cdot \boldsymbol{H} + (r \cdot \boldsymbol{v} + s \cdot \boldsymbol{u}) \cdot G + \delta \cdot H) \\ &+ (r \cdot s) \cdot G + \eta \cdot H \\ &= (r + \boldsymbol{u} \cdot e) \cdot e \cdot \boldsymbol{G} + (s + \boldsymbol{v} \cdot e) \cdot e \cdot \boldsymbol{H} \\ &+ (r + \boldsymbol{u} \cdot e) \cdot (s + \boldsymbol{v} \cdot e) \cdot G + (\eta + \delta \cdot e + \alpha \cdot e^2) \cdot H \\ &= (r' \cdot e) \cdot \boldsymbol{G} + (s' \cdot e) \cdot \boldsymbol{H} + (r' \cdot s') \cdot G + \delta' \cdot H \;. \end{split}$$

Assume now that the argument system Π_{mIP} has perfect completeness for instances of dimension $\hat{d} = d/2 = 2^{t-1}$, and we prove that it has perfect completeness also for instances of dimension $d = 2^t$. Let $(\boldsymbol{G}, \boldsymbol{H}, G, H, P) \in \mathbb{G}^{2d+3}$ and $(\boldsymbol{u}, \boldsymbol{v}, \alpha) \in \mathbb{Z}_q^{2d+1}$ be an instance and a witness, respectively, such that $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{\text{mIP}}$. That is,

$$P = \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot G + \alpha \cdot H .$$

We show that, for any values $d_L, d_R \leftarrow \mathbb{Z}_q$ sampled by the prover and for any value $e \leftarrow \mathbb{Z}_q^*$ sampled by the verifier, it holds that $((\hat{\boldsymbol{G}}, \hat{\boldsymbol{H}}, G, H, \hat{P}), (\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}, \hat{\boldsymbol{\alpha}})) \in \mathcal{R}_{\text{mIP}}$, where

$$c_{L} = \langle \boldsymbol{u}_{1}, \boldsymbol{v}_{2} \rangle \in \mathbb{Z}_{q}$$

$$c_{R} = \langle \boldsymbol{u}_{2}, \boldsymbol{v}_{1} \rangle \in \mathbb{Z}_{q}$$

$$L = \langle \boldsymbol{u}_{1}, \boldsymbol{G}_{2} \rangle + \langle \boldsymbol{v}_{2}, \boldsymbol{H}_{1} \rangle + c_{L} \cdot \boldsymbol{G} + d_{L} \cdot \boldsymbol{H} \in \mathbb{G}$$

$$R = \langle \boldsymbol{u}_{2}, \boldsymbol{G}_{1} \rangle + \langle \boldsymbol{v}_{1}, \boldsymbol{H}_{2} \rangle + c_{R} \cdot \boldsymbol{G} + d_{R} \cdot \boldsymbol{H} \in \mathbb{G}$$

$$\hat{\boldsymbol{G}} = e^{-1} \cdot \boldsymbol{G}_{1} + e \cdot \boldsymbol{G}_{2} \in \mathbb{G}^{\hat{d}}$$

$$\hat{\boldsymbol{H}} = e \cdot \boldsymbol{H}_{1} + e^{-1} \cdot \boldsymbol{H}_{2} \in \mathbb{G}^{\hat{d}}$$

$$\hat{\boldsymbol{P}} = e^{2} \cdot \boldsymbol{L} + \boldsymbol{P} + e^{-2} \cdot \boldsymbol{R} \in \mathbb{G}$$

$$\hat{\boldsymbol{u}} = e \cdot \boldsymbol{u}_{1} + e^{-1} \cdot \boldsymbol{u}_{2} \in \mathbb{Z}_{q}^{\hat{d}}$$

$$\hat{\boldsymbol{v}} = e^{-1} \cdot \boldsymbol{v}_{1} + e \cdot \boldsymbol{v}_{2} \in \mathbb{Z}_{q}^{\hat{d}}$$

$$\hat{\boldsymbol{a}} = e^{2} \cdot d_{L} + \alpha + e^{-2} \cdot d_{R} \in \mathbb{Z}_{q}$$

are the values computed by the prover and verifier as instructed by the protocol. The perfect completeness of the argument system for instances of dimension $d/2 = 2^{t-1}$ then guarantees that the verifier would always accept when invoking the protocol with the instance $(\hat{\boldsymbol{G}}, \hat{\boldsymbol{H}}, G, H, \hat{P})$ whose dimension is d/2. In order to show that $((\hat{\boldsymbol{G}}, \hat{\boldsymbol{H}}, G, H, \hat{P}), (\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}, \hat{\alpha})) \in \mathcal{R}_{\text{mIP}}$, we have to show that

$$\hat{P} = \langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{G}} \rangle + \langle \hat{\boldsymbol{v}}, \hat{\boldsymbol{H}} \rangle + \langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}} \rangle \cdot G + \hat{\alpha} \cdot H .$$

First, for the inner-product $\langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}} \rangle$ it holds that

$$\langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}} \rangle = \langle e \cdot \boldsymbol{u}_1 + e^{-1} \cdot \boldsymbol{u}_2, e^{-1} \cdot \boldsymbol{v}_1 + e \cdot \boldsymbol{v}_2 \rangle$$

$$= e^2 \cdot \langle \boldsymbol{u}_1, \boldsymbol{v}_2 \rangle + \langle \boldsymbol{u}_1, \boldsymbol{v}_1 \rangle + \langle \boldsymbol{u}_2, \boldsymbol{v}_2 \rangle + e^{-2} \cdot \langle \boldsymbol{u}_2, \boldsymbol{v}_1 \rangle$$

$$= e^2 \cdot c_L + \langle \boldsymbol{u}, \boldsymbol{v} \rangle + e^{-2} \cdot c_R .$$

Similarly, for the inner-products $\langle \hat{\pmb{u}}, \hat{\pmb{G}} \rangle$ and $\langle \hat{\pmb{v}}, \hat{\pmb{H}} \rangle$ it holds that

$$\langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{G}} \rangle = \langle e \cdot \boldsymbol{u}_1 + e^{-1} \cdot \boldsymbol{u}_2, e^{-1} \cdot \boldsymbol{G}_1 + e \cdot \boldsymbol{G}_2 \rangle$$

$$= e^2 \cdot \langle \boldsymbol{u}_1, \boldsymbol{G}_2 \rangle + \langle \boldsymbol{u}_1, \boldsymbol{G}_1 \rangle + \langle \boldsymbol{u}_2, \boldsymbol{G}_2 \rangle + e^{-2} \cdot \langle \boldsymbol{u}_2, \boldsymbol{G}_1 \rangle$$

$$= e^2 \cdot \langle \boldsymbol{u}_1, \boldsymbol{G}_2 \rangle + \langle \boldsymbol{u}, \boldsymbol{G} \rangle + e^{-2} \cdot \langle \boldsymbol{u}_2, \boldsymbol{G}_1 \rangle$$

and

$$\langle \hat{\boldsymbol{v}}, \hat{\boldsymbol{H}} \rangle = \langle e^{-1} \cdot \boldsymbol{v}_1 + e \cdot \boldsymbol{v}_2, e \cdot \boldsymbol{H}_1 + e^{-1} \cdot \boldsymbol{H}_2 \rangle$$

$$= e^2 \cdot \langle \boldsymbol{v}_2, \boldsymbol{H}_1 \rangle + \langle \boldsymbol{v}_1, \boldsymbol{H}_1 \rangle + \langle \boldsymbol{v}_2, \boldsymbol{H}_2 \rangle + e^{-2} \cdot \langle \boldsymbol{v}_1, \boldsymbol{H}_2 \rangle$$

$$= e^2 \cdot \langle \boldsymbol{v}_2, \boldsymbol{H}_1 \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + e^{-2} \cdot \langle \boldsymbol{v}_1, \boldsymbol{H}_2 \rangle .$$

These imply that for $\hat{P} = e^2 \cdot L + P + e^{-2} \cdot R$ it holds that

$$\hat{P} = e^{2} \cdot L + P + e^{-2} \cdot R
= e^{2} \cdot (\langle \boldsymbol{u}_{1}, \boldsymbol{G}_{2} \rangle + \langle \boldsymbol{v}_{2}, \boldsymbol{H}_{1} \rangle + c_{L} \cdot G + d_{L} \cdot H)
+ \langle \boldsymbol{u}, \boldsymbol{G} \rangle + \langle \boldsymbol{v}, \boldsymbol{H} \rangle + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \cdot G + \alpha \cdot H
+ e^{-2} \cdot (\langle \boldsymbol{u}_{2}, \boldsymbol{G}_{1} \rangle + \langle \boldsymbol{v}_{1}, \boldsymbol{H}_{2} \rangle + c_{R} \cdot G + d_{R} \cdot H)
= \langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{G}} \rangle + \langle \hat{\boldsymbol{v}}, \hat{\boldsymbol{H}} \rangle + \langle \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}} \rangle \cdot G + \hat{\alpha} \cdot H$$

as required.

Claim B.3. The argument system Π_{mIP} has perfect special honest-verifier zero-knowledge.

Proof. We prove via induction that the argument system Π_{mIP} has perfect special honest-verifier zero-knowledge for instances of dimension $d=2^t$ for any integer $t \geq 0$. For t=0 (and thus d=1), let \mathcal{S}_1 be the simulator that is defined as follows on input $((\mathbf{G}, \mathbf{H}, G, H, P), \rho)$:

- 1. Parse $\rho = e$, where $e \in \mathbb{Z}_q^*$ is the verifier's randomness.
- 2. Uniformly sample $A \leftarrow \mathbb{G}$ and $r', s', \delta' \leftarrow \mathbb{Z}_q$.
- 3. Compute $B = -e^2 \cdot P e \cdot A + (r' \cdot e) \cdot G + (s' \cdot e) \cdot H + (r' \cdot s') \cdot G + \delta' \cdot H \in \mathbb{G}$.
- 4. Output $(A, B, e, r', s', \delta')$.

Let A_1 and A_2 be any two algorithms as in Definition 2.3, and fix any triplet $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha), \rho)$ produced by A_1 such that $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{mIP}$. We need to show that, conditioned on $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha), \rho)$, the distribution of the transcript produced by the simulator S_1 is identical to the distribution of an honestly-generated transcript (thus, A_2 will have no advantage in distinguishing the two cases – as required by Definition 2.3).

First, in both cases, the value e is uniquely determined by the verifier's randomness $\rho = e$. Second, in both cases, the group element B is computed as the same deterministic function of A, r', s' and δ' . Specifically, for an honestly-generated transcript, the fact that $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{mIP}$ (together with the perfect completeness of the argument system) guarantees that verifier accepts and thus

$$e^2 \cdot P + e \cdot A + B = (r' \cdot e) \cdot G + (s' \cdot e) \cdot H + (r' \cdot s') \cdot G + \delta' \cdot H$$
.

This is equivalent to letting

$$B = -e^2 \cdot P - e \cdot A + (r' \cdot e) \cdot \mathbf{G} + (s' \cdot e) \cdot \mathbf{H} + (r' \cdot s') \cdot G + \delta' \cdot H ,$$

which is the identical to the computation of the group element B by the simulator S_1 . Therefore, in order to prove that the distribution of the transcript produced by the simulator S_1 is identical to the distribution of an honestly-generated transcript, it suffices to consider the distribution of (A, r', s', δ') given $((G, H, G, H, P), (u, v, \alpha), \rho)$.

Note that, in the transcript produced by S_1 , by the description of S_1 it holds that (A, r', s', δ') is distributed uniformly given $((G, H, G, H, P), (u, v, \alpha), \rho)$. Therefore, we need to prove that this holds also in an honestly-generated transcript. We prove this by letting $A = \zeta_A \cdot H$ for $\zeta_A \in \mathbb{Z}_q$, and proving that the vector $(\zeta_A, r', s', \delta')$ is obtained by applying an invertible transformation to the vector (η, r, s, δ) – which is uniformly distributed in an honestly-generated transcript (the invertible transformation may depend on $((G, H, G, H, P), (u, v, \alpha), \rho)$). Letting $G = \zeta_G \cdot H$, $H = \zeta_H \cdot H$ and $G = \zeta_G \cdot H$ for $\zeta_G, \zeta_H, \zeta_G \in \mathbb{Z}_q$, it holds that

$$\begin{bmatrix} \zeta_A \\ r' \\ s' \\ \delta' \end{bmatrix} = \begin{bmatrix} 0 & \zeta_G + \zeta_G \cdot \boldsymbol{v} & \zeta_H + \zeta_G \cdot \boldsymbol{u} & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & e \end{bmatrix} \cdot \begin{bmatrix} \eta \\ r \\ s \\ \delta \end{bmatrix} + \begin{bmatrix} 0 \\ \boldsymbol{u} \cdot e \\ \boldsymbol{v} \cdot e \\ \alpha \cdot e^2 \end{bmatrix}$$

The above 4×4 matrix is invertible for any given $\zeta_{\mathbf{G}}, \zeta_{\mathbf{H}}, \zeta_{G}, \mathbf{u}, \mathbf{v}, \alpha, e \in \mathbb{Z}_{q}$, and this concludes the case d = 1.

Assume now that the argument system Π_{mIP} has perfect special honest-verifier zero-knowledge for instances of dimension $\hat{d} = d/2 = 2^{t-1}$ (provided by a simulator \mathcal{S}_{t-1}), and we prove that it has perfect special honest-verifier zero-knowledge also for instances of dimension $d = 2^t$. Let \mathcal{S}_t be the simulator that is defined as follows on input $((G, H, G, H, P), \rho)$:

- 1. Parse $\rho = (e, \rho_{t-1})$, where $e \in \mathbb{Z}_q^*$ is the first challenge sent by the verifier.
- 2. Uniformly sample $L, R \leftarrow \mathbb{G}$.
- 3. Let $G = (G_1, G_2) \in \mathbb{G}^{\hat{d}} \times \mathbb{G}^{\hat{d}}$ and $H = (H_1, H_2) \in \mathbb{G}^{\hat{d}} \times \mathbb{G}^{\hat{d}}$, and compute

$$\hat{\mathbf{G}} = e^{-1} \cdot \mathbf{G}_1 + e \cdot \mathbf{G}_2 \in \mathbb{G}^{\hat{d}}$$

$$\hat{\mathbf{H}} = e \cdot \mathbf{H}_1 + e^{-1} \cdot \mathbf{H}_2 \in \mathbb{G}^{\hat{d}}$$

$$\hat{P} = e^2 \cdot L + P + e^{-2} \cdot R \in \mathbb{G}.$$

- 4. Invoke the simulator S_{t-1} on the input $((\hat{G}, \hat{H}, G, H, \hat{P}), \rho_{t-1}))$ to obtain a transcript tr_{t-1} .
- 5. Output $tr_t = (L, R, e, tr_{t-1})$.

Let A_1 and A_2 be any two algorithms as in Definition 2.3, and fix any triplet $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha), \rho)$ produced by A_1 such that $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{mIP}$. We need to show that, conditioned on $((\boldsymbol{G}, \boldsymbol{H}, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha), \rho)$, the distribution of the transcript produced by the simulator S_t is identical to the distribution of an honestly-generated transcript (thus, A_2 will have no advantage in distinguishing the two cases – as required by Definition 2.3).

First, in both cases, the value e is uniquely determined by the verifier's randomness $\rho = (e, \rho_{t-1})$. Second, in both cases, the group elements L and R are independently and uniformly distributed given $((G, H, G, H, P), (\mathbf{u}, \mathbf{v}, \alpha), \rho)$. Specifically, for the transcript produced by \mathcal{S}_t , this follows directly from the fact that \mathcal{S}_t samples L and R independently and uniformly. For an honestly-generated transcript this follows from their randomizing terms $d_L \cdot H$ and $d_R \cdot H$, where $d_L, d_R \in \mathbb{Z}_q$ are independently and uniformly distributed. Finally, in both cases the instance $(\hat{G}, \hat{H}, G, H, \hat{P})$ is computed as the same deterministic function of $((G, H, G, H, P), (\mathbf{u}, \mathbf{v}, \alpha), \rho)$, L, R and e. The perfect completeness of the argument system guarantees that there exists a witness $(\hat{\mathbf{u}}, \hat{\mathbf{v}}, \hat{\alpha})$ such that $((\hat{G}, \hat{H}, G, H, \hat{P}), (\hat{\mathbf{u}}, \hat{\mathbf{v}}, \hat{\alpha})) \in \mathcal{R}_{\text{mIP}}$, and therefore the perfect special honest-verifier zero-knowledge for instances of dimension $\hat{d} = d/2 = 2^{t-1}$ guarantees that the transcript tr_{t-1} produced by \mathcal{S}_{t-1} is distributed identically to an honestly-generated one conditioned on all previous values.

Claim B.4. There exists a probabilistic polynomial-time algorithm Ext that, on input any \mathcal{R}_{mIP} instance (G, H, G, H, P) of dimension $d = 2^t \geq 1$ together with any corresponding $(4, \ldots, 4, 5)$ -transcript tree of depth $\log_2 d + 1$ for the argument system Π_{mIP} , produces either a witness $(\boldsymbol{u}, \boldsymbol{v}, \alpha)$ such that $((G, H, G, H, P), (\boldsymbol{u}, \boldsymbol{v}, \alpha)) \in \mathcal{R}_{mIP}$ or a non-trivial discrete-logarithm relation for (G, H, G, H).

Proof. We prove via induction that for any dimension $d = 2^t$, where $t \ge 0$, there exists a probabilistic polynomial-time algorithm Ext_t that, on input any $\mathcal{R}_{\mathsf{mIP}}$ instance (G, H, G, H, P) of dimension d together with any corresponding $(4, \ldots, 4, 5)$ -transcript tree of depth t+1, produces either a witness (u, v, α) such that $((G, H, G, H, P), (u, v, \alpha)) \in \mathcal{R}_{\mathsf{mIP}}$ or a non-trivial discrete-logarithm relation for (G, H, G, H).

For t=0 (and thus d=1), a 5-transcript tree of depth 1 for an instance (G, H, G, H, P) consists of group elements $A, B \in \mathbb{G}$ together with 5 distinct challenges $e_1, \ldots, e_5 \in \mathbb{Z}_q^*$ and corresponding triplets $(r'_1, s'_1, \delta'_1), \ldots, (r'_5, s'_5, \delta'_5) \in \mathbb{Z}_q^3$ such that

$$e_i^2 \cdot P + e_i \cdot A + B = r_i' e_i \cdot \mathbf{G} + s_i' e_i \cdot \mathbf{H} + r_i' s_i' \cdot G + \delta_i' \cdot H$$
(B.1)

for every $i \in [5]$. Since the e_i 's are distinct, then the matrix $M \in \mathbb{Z}_q^{3\times 3}$ defined as

$$M = \begin{bmatrix} e_1^2 & e_1 & 1\\ e_2^2 & e_2 & 1\\ e_3^2 & e_3 & 1 \end{bmatrix}$$

is invertible. By efficiently computing its inverse M^{-1} , and then multiplying it by the three equations obtained from Eq. (B.1) for i=1,2,3 we obtain $p_{\mathbf{G}},p_{\mathbf{H}},p_{G},p_{H},a_{\mathbf{G}},a_{\mathbf{H}},a_{G},a_{H},b_{\mathbf{G}},b_{\mathbf{H}},b_{G},b_{H} \in \mathbb{Z}_{q}$ such that

$$P = p_{\mathbf{G}} \cdot \mathbf{G} + p_{\mathbf{H}} \cdot \mathbf{H} + p_{G} \cdot G + p_{H} \cdot H$$

$$A = a_{\mathbf{G}} \cdot \mathbf{G} + a_{\mathbf{H}} \cdot \mathbf{H} + a_{G} \cdot G + a_{H} \cdot H$$

$$B = b_{\mathbf{G}} \cdot \mathbf{G} + b_{\mathbf{H}} \cdot \mathbf{H} + b_{G} \cdot G + b_{H} \cdot H .$$
(B.2)

Combining this representation of P, A and B with Eq. (B.1), for every $i \in [5]$ it holds that

$$0 = (e_i^2 \cdot p_{\mathbf{G}} + e_i \cdot a_{\mathbf{G}} + b_{\mathbf{G}} - r_i' \cdot e_i) \cdot \mathbf{G} + (e_i^2 \cdot p_{\mathbf{H}} + e_i \cdot a_{\mathbf{H}} + b_{\mathbf{H}} - s_i' \cdot e_i) \cdot \mathbf{H}$$

$$+ (e_i^2 \cdot p_G + e_i \cdot a_G + b_G - r_i' \cdot s_i') \cdot \mathbf{G} + (e_i^2 \cdot p_H + e_i \cdot a_H + b_H - \delta_i') \cdot \mathbf{H} .$$

This implies that

$$0 = e_i^2 \cdot p_G + e_i \cdot a_G + b_G - r_i' \cdot e_i$$
(B.3)

$$0 = e_i^2 \cdot p_{\mathbf{H}} + e_i \cdot a_{\mathbf{H}} + b_{\mathbf{H}} - s_i' \cdot e_i \tag{B.4}$$

$$0 = e_i^2 \cdot p_G + e_i \cdot a_G + b_G - r_i' \cdot s_i'$$

$$0 = e_i^2 \cdot p_H + e_i \cdot a_H + b_H - \delta_i'$$
(B.5)

as otherwise we obtain a non-trivial discrete-logarithm relation for (G, H, G, H). From Eq. (B.3) and (B.4) we obtain

$$r'_i = e_i \cdot p_G + a_G + e_i^{-1} \cdot b_G$$

$$s'_i = e_i \cdot p_H + a_H + e_i^{-1} \cdot b_H$$

and combining these with Eq. (B.5) we obtain

$$e_i^2 \cdot p_G + e_i \cdot a_G + b_G = r_i' \cdot s_i'$$

$$= (e_i \cdot p_G + a_G + e_i^{-1} \cdot b_G) \cdot (e_i \cdot p_H + a_H + e_i^{-1} \cdot b_H)$$

$$= (b_G \cdot b_H) \cdot e_i^{-2} + (b_G \cdot a_H + a_G \cdot b_H) \cdot e_i^{-1}$$

$$+ (p_G \cdot b_H + a_G \cdot a_H + b_G \cdot p_H)$$

$$+ (p_G \cdot a_H + a_G \cdot p_H) \cdot e_i + (p_G \cdot p_H) \cdot e_i^2.$$

Therefore, for every $i \in [5]$ it holds that

$$0 = (b_{\mathbf{G}} \cdot b_{\mathbf{H}}) \cdot e_{i}^{-2} + (b_{\mathbf{G}} \cdot a_{\mathbf{H}} + a_{\mathbf{G}} \cdot b_{\mathbf{H}}) \cdot e_{i}^{-1}$$
$$+ (p_{\mathbf{G}} \cdot b_{\mathbf{H}} + a_{\mathbf{G}} \cdot a_{\mathbf{H}} + b_{\mathbf{G}} \cdot p_{\mathbf{H}} - b_{\mathbf{G}})$$
$$+ (p_{\mathbf{G}} \cdot a_{\mathbf{H}} + a_{\mathbf{G}} \cdot p_{\mathbf{H}} - a_{\mathbf{G}}) \cdot e_{i} + (p_{\mathbf{G}} \cdot p_{\mathbf{H}} - p_{\mathbf{G}}) \cdot e_{i}^{2}.$$

Since the e_i 's are distinct then the vectors $\{(e_i^2, e_i, 1, e_i^{-1}, e_i^{-2})\}_{i \in [5]}$ are linearly independent over \mathbb{Z}_q , and therefore $p_G = p_G \cdot p_H$. Thus, letting $\boldsymbol{u} = p_G \in \mathbb{Z}_q$, $\boldsymbol{v} = p_H \in \mathbb{Z}_q$ and $\alpha = p_H \in \mathbb{Z}_q$, from Eq. (B.2) we obtain

$$P = p_{G} \cdot G + p_{H} \cdot H + p_{G} \cdot G + p_{H} \cdot H$$

= $\langle u, G \rangle + \langle v, H \rangle + \langle u, v \rangle \cdot G + \alpha \cdot H$

as required.

Assume now that there exists a probabilistic polynomial-time algorithm Ext_{t-1} that, on input any $\mathcal{R}_{\mathrm{mIP}}$ instance (G, H, G, H, P) of dimension $\hat{d} = d/2 = 2^{t-1}$ together with any corresponding $(4, \ldots, 4, 5)$ -transcript tree of depth t, produces either a witness (u, v, α) such that $((G, H, G, H, P), (u, v, \alpha)) \in \mathcal{R}_{\mathrm{mIP}}$ or a non-trivial discrete-logarithm relation for (G, H, G, H). We prove that there exists exists a probabilistic polynomial-time algorithm Ext_t that, on input any $\mathcal{R}_{\mathrm{mIP}}$ instance (G, H, G, H, P) of dimension $d = 2^t$ together with any corresponding $(4, \ldots, 4, 5)$ -transcript tree of depth t+1, produces either a witness (u, v, α) such that $((G, H, G, H, P), (u, v, \alpha)) \in \mathcal{R}_{\mathrm{mIP}}$ or a non-trivial discrete-logarithm relation for (G, H, G, H).

Note that a $(4,\ldots,4,5)$ -transcript tree of depth t+1 for an instance $(\boldsymbol{G},\boldsymbol{H},G,H,P)$ of dimension $d=2^t$ consists of group elements $L,R\in\mathbb{G}$ together with 4 first-level nodes corresponding to challenges $e_1,\ldots,e_4\in\mathbb{Z}_q^*$ such that $e_1^2,\ldots,e_4^2\in\mathbb{Z}_q^*$ are distinct (recall, as discussed in Section 2.2, that we consider a refinement to the notion of a transcript tree in which the squares of the challenges are distinct). Each such challenge defines an instance $(\hat{\boldsymbol{G}}_i,\hat{\boldsymbol{H}}_i,G,H,\hat{P}_i)$ of dimension $\hat{d}=d/2=2^{t-1}$, where

$$\hat{\boldsymbol{G}}_{i} = e_{i}^{-1} \cdot \boldsymbol{G}_{1} + e_{i} \cdot \boldsymbol{G}_{2} \in \mathbb{G}^{\hat{d}}$$

$$\hat{\boldsymbol{H}}_{i} = e_{i} \cdot \boldsymbol{H}_{1} + e_{i}^{-1} \cdot \boldsymbol{H}_{2} \in \mathbb{G}^{\hat{d}}$$

$$\hat{P}_{i} = e_{i}^{2} \cdot L + P + e_{i}^{-2} \cdot R \in \mathbb{G}$$

$$\boldsymbol{G} = (\boldsymbol{G}_{1}, \boldsymbol{G}_{2}) \in \mathbb{G}^{\hat{d}} \times \mathbb{G}^{\hat{d}}$$

$$\boldsymbol{H} = (\boldsymbol{H}_{1}, \boldsymbol{H}_{2}) \in \mathbb{G}^{\hat{d}} \times \mathbb{G}^{\hat{d}}.$$

For each such instance, the corresponding first-level node serves as the root of a $(4, \ldots, 4, 5)$ -transcript sub-tree of depth t. The extractor Ext_t first invokes the extractor Ext_{t-1} on each of the 4 sub-trees, and for each $i \in [4]$ obtains either a witness $(\hat{\boldsymbol{u}}_i, \hat{\boldsymbol{v}}_i, \hat{\alpha}_i) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q$ such that $((\hat{\boldsymbol{G}}_i, \hat{\boldsymbol{H}}_i, G, H, \hat{P}_i), (\hat{\boldsymbol{u}}_i, \hat{\boldsymbol{v}}_i, \hat{\alpha}_i)) \in \mathcal{R}_{\text{mIP}}$, or a non-trivial discrete-logarithm relation for $(\hat{\boldsymbol{G}}_i, \hat{\boldsymbol{H}}_i, G, H)$. Any such relation yields a corresponding relation for $(\boldsymbol{G}, \boldsymbol{H}, G, H)$, and therefore for the remainder of the proof we assume that the extractor obtains a witness $(\hat{\boldsymbol{u}}_i, \hat{\boldsymbol{v}}_i, \hat{\alpha}_i)$ for each $i \in [4]$. That is, it holds that

$$e_{i}^{2} \cdot L + P + e_{i}^{-2} \cdot R = \hat{P}_{i}$$

$$= \langle \hat{\boldsymbol{u}}_{i}, \hat{\boldsymbol{G}}_{i} \rangle + \langle \hat{\boldsymbol{v}}_{i}, \hat{\boldsymbol{H}}_{i} \rangle + \langle \hat{\boldsymbol{u}}_{i}, \hat{\boldsymbol{v}}_{i} \rangle \cdot G + \hat{\alpha}_{i} \cdot H$$

$$= \langle (e_{i}^{-1} \cdot \hat{\boldsymbol{u}}_{i} || e_{i} \cdot \hat{\boldsymbol{u}}_{i}), \boldsymbol{G} \rangle + \langle (e_{i} \cdot \hat{\boldsymbol{v}}_{i} || e_{i}^{-1} \cdot \hat{\boldsymbol{v}}_{i}), \boldsymbol{H} \rangle$$

$$+ \langle \hat{\boldsymbol{u}}_{i}, \hat{\boldsymbol{v}}_{i} \rangle \cdot G + \hat{\alpha}_{i} \cdot H$$
(B.6)

for each $i \in [4]$. Since the e_i^2 's are distinct, then the matrix $M \in \mathbb{Z}_q^{3 \times 3}$ defined as

$$M = \begin{bmatrix} e_1^2 & 1 & e_1^{-2} \\ e_2^2 & 1 & e_2^{-2} \\ e_3^2 & 1 & e_3^{-2} \end{bmatrix}$$

is invertible. As in the case d=1, this enables to efficiently compute $p_{\mathbf{G}}, p_{\mathbf{H}}, \ell_{\mathbf{G}}, \ell_{\mathbf{H}}, r_{\mathbf{G}}, r_{\mathbf{H}} \in \mathbb{Z}_q^d$ and $p_G, p_H, \ell_G, \ell_H, r_G, r_H \in \mathbb{Z}_q$ such that

$$L = \langle \ell_{\mathbf{G}}, \mathbf{G} \rangle + \langle \ell_{\mathbf{H}}, \mathbf{H} \rangle + \ell_{\mathbf{G}} \cdot G + \ell_{\mathbf{H}} \cdot H$$
(B.7)

$$P = \langle p_{G}, G \rangle + \langle p_{H}, H \rangle + p_{G} \cdot G + p_{H} \cdot H$$
(B.8)

$$R = \langle r_{\mathbf{G}}, \mathbf{G} \rangle + \langle r_{\mathbf{H}}, \mathbf{H} \rangle + r_{G} \cdot G + r_{H} \cdot H .$$
(B.9)

Letting $u = p_G$, $v = p_H$ and $\alpha = p_H$, we are left to prove that $p_G = \langle p_G, p_H \rangle$, and then Eq. (B.8) would establish that $((G, H, G, H, P), (u, v, \alpha)) \in \mathcal{R}_{mIP}$ as required.

Combining the representation of P, L and R provided by Eq. (B.7)–(B.9) with Eq. (B.6), for every $i \in [4]$ it holds that

$$0 = \langle e_i^2 \cdot \ell_{\mathbf{G}} + p_{\mathbf{G}} + e_i^{-2} \cdot r_{\mathbf{G}} - (e_i^{-1} \cdot \hat{\mathbf{u}}_i || e_i \cdot \hat{\mathbf{u}}_i), \mathbf{G} \rangle$$

$$+ \langle e_i^2 \cdot \ell_{\mathbf{H}} + p_{\mathbf{H}} + e_i^{-2} \cdot r_{\mathbf{H}} - (e_i \cdot \hat{\mathbf{v}}_i || e_i^{-1} \cdot \hat{\mathbf{v}}_i), \mathbf{H} \rangle$$

$$+ (e_i^2 \cdot \ell_G + p_G + e_i^{-2} \cdot r_G - \langle \hat{\mathbf{u}}_i, \hat{\mathbf{v}}_i \rangle) \cdot G$$

$$+ (e_i^2 \cdot \ell_H + p_H + e_i^{-2} \cdot r_H - \hat{\alpha}_i) \cdot H.$$

which implies

$$0^{d} = e_{i}^{2} \cdot \ell_{G} + p_{G} + e_{i}^{-2} \cdot r_{G} - (e_{i}^{-1} \cdot \hat{\boldsymbol{u}}_{i} || e_{i} \cdot \hat{\boldsymbol{u}}_{i})$$
(B.10)

$$0^{d} = e_{i}^{2} \cdot \ell_{\mathbf{H}} + p_{\mathbf{H}} + e_{i}^{-2} \cdot r_{\mathbf{H}} - (e_{i} \cdot \hat{\mathbf{v}}_{i} || e_{i}^{-1} \cdot \hat{\mathbf{v}}_{i})$$
(B.11)

$$0 = e_i^2 \cdot \ell_G + p_G + e_i^{-2} \cdot r_G - \langle \hat{\boldsymbol{u}}_i, \hat{\boldsymbol{v}}_i \rangle$$

$$0 = e_i^2 \cdot \ell_H + p_H + e_i^{-2} \cdot r_H - \hat{\alpha}_i.$$
(B.12)

as otherwise we obtain a non-trivial discrete-logarithm relation for (G, H, G, H). Letting $\ell_G = (\ell_{G,1}, \ell_{G,2}) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}, \ p_G = (p_{G,1}, p_{G,2}) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}$ and $r_G = (r_{G,1}, r_{G,2}) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}$, Eq. (B.10) can be rewritten as

$$e_i^2 \cdot \ell_{G,1} + p_{G,1} + e_i^{-2} \cdot r_{G,1} = e_i^{-1} \cdot \hat{\boldsymbol{u}}_i$$

$$e_i^2 \cdot \ell_{G,2} + p_{G,2} + e_i^{-2} \cdot r_{G,2} = e_i \cdot \hat{\boldsymbol{u}}_i .$$
(B.13)

This implies that

$$e_i^3 \cdot \ell_{G,1} + e_i \cdot p_{G,1} + e_i^{-1} \cdot r_{G,1} = \hat{\boldsymbol{u}}_i = e_i \cdot \ell_{G,2} + e_i^{-1} \cdot p_{G,2} + e_i^{-3} \cdot r_{G,2}$$

which yields

$$0^{\hat{d}} = e_i^3 \cdot \ell_{G,1} + e_i \cdot (p_{G,1} - \ell_{G,2}) + e_i^{-1} \cdot (r_{G,1} - p_{G,2}) - e_i^{-3} \cdot r_{G,2}$$

for each $i \in [4]$. Since the e_i^2 's are distinct then the vectors $\{(e_i^3, e_i, e_i^{-1}, e_i^{-3})\}_{i \in [4]}$ are linearly independent over \mathbb{Z}_q , and therefore

$$\ell_{G,1} = 0^{\hat{d}}, \quad p_{G,1} = \ell_{G,2}, \quad r_{G,1} = p_{G,2}, \quad r_{G,2} = 0^{\hat{d}}.$$

Eq. (B.13) now implies that for each $i \in [4]$ it holds that

$$\hat{\boldsymbol{u}}_{i} = e_{i}^{3} \cdot \ell_{G,1} + e_{i} \cdot p_{G,1} + e_{i}^{-1} \cdot r_{G,1}$$

$$= e_{i} \cdot p_{G,1} + e_{i}^{-1} \cdot p_{G,2} . \tag{B.14}$$

Similarly, letting $\ell_{\boldsymbol{H}} = (\ell_{\boldsymbol{H},1}, \ell_{\boldsymbol{H},2}) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}, p_{\boldsymbol{H}} = (p_{\boldsymbol{H},1}, p_{\boldsymbol{H},2}) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}} \text{ and } r_{\boldsymbol{H}} = (r_{\boldsymbol{H},1}, r_{\boldsymbol{H},2}) \in \mathbb{Z}_q^{\hat{d}} \times \mathbb{Z}_q^{\hat{d}}, \text{ Eq. (B.11) can be rewritten as}$

$$e_{i}^{2} \cdot \ell_{H,1} + p_{H,1} + e_{i}^{-2} \cdot r_{H,1} = e_{i} \cdot \hat{\boldsymbol{v}}_{i}$$

$$e_{i}^{2} \cdot \ell_{H,2} + p_{H,2} + e_{i}^{-2} \cdot r_{H,2} = e_{i}^{-1} \cdot \hat{\boldsymbol{v}}_{i} .$$
(B.15)

This implies that

$$e_i \cdot \ell_{H,1} + e_i^{-1} \cdot p_{H,1} + e_i^{-3} \cdot r_{H,1} = \hat{\boldsymbol{v}}_i = e_i^3 \cdot \ell_{H,2} + e_i \cdot p_{H,2} + e_i^{-1} \cdot r_{H,2}$$

which yields

$$0 = e_i^3 \cdot \ell_{H,2} + e_i \cdot (p_{H,2} - \ell_{H,1}) + e_i^{-1} \cdot (r_{H,2} - p_{H,1}) - e_i^{-3} \cdot r_{H,1}$$

for each $i \in [4]$. Since the e_i^2 's are distinct then the vectors $\{(e_i^3, e_i, e_i^{-1}, e_i^{-3})\}_{i \in [4]}$ are linearly independent over \mathbb{Z}_q , and therefore

$$\ell_{H,2} = 0^{\hat{d}}, \quad p_{H,2} = \ell_{H,1}, \quad r_{H,2} = p_{H,1}, \quad r_{H,1} = 0^{\hat{d}}.$$

Eq. (B.15) now implies that for each $i \in [4]$ it holds that

$$\hat{\mathbf{v}}_{i} = e_{i} \cdot \ell_{H,1} + e_{i}^{-1} \cdot p_{H,1} + e_{i}^{-3} \cdot r_{H,1}$$

$$= e_{i}^{-1} \cdot p_{H,1} + e_{i} \cdot p_{H,2} . \tag{B.16}$$

Considering the inner-product $\langle \hat{\boldsymbol{u}}_i, \hat{\boldsymbol{v}}_i \rangle$, by combining Eq. (B.12) with Eq. (B.14) and Eq. (B.16) we obtain

$$e_{i}^{2} \cdot \ell_{G} + p_{G} + e_{i}^{-2} \cdot r_{G} = \langle \hat{\boldsymbol{u}}_{i}, \hat{\boldsymbol{v}}_{i} \rangle$$

$$= \langle e_{i} \cdot p_{G,1} + e_{i}^{-1} \cdot p_{G,2}, e_{i}^{-1} \cdot p_{H,1} + e_{i} \cdot p_{H,2} \rangle$$

$$= e_{i}^{-2} \cdot \langle p_{G,2}, p_{H,1} \rangle + \langle p_{G}, p_{H} \rangle + e_{i}^{2} \cdot \langle p_{G,1}, p_{H,2} \rangle$$

and therefore

$$0 = e_i^{-2} \cdot (\langle p_{G,2}, p_{H,1} \rangle - r_G) + (\langle p_G, p_H \rangle - p_G) + e_i^2 \cdot (\langle p_{G,1}, p_{H,2} \rangle - \ell_G).$$

Since the e_i^2 's are distinct then the vectors $\{(e_i^{-2}, 1, e_i^2)\}_{i \in [3]}$ are linearly independent over \mathbb{Z}_q , and therefore $p_G = \langle p_G, p_H \rangle$ as required.

C An Auxiliary Claim

Definition C.1. Let $n_{\alpha}, n_{\beta}, n_{\gamma} \geq 1$. A set of triplets $\{(\alpha_i, \beta_{i,j}, \gamma_{i,j,k})\}_{i \in [n_{\alpha}], j \in [n_{\beta}], k \in [n_{\gamma}]}$ is path distinct if the following hold:

- 1. The values $\{\alpha_i\}_{i\in[n_\alpha]}$ are distinct.
- 2. For every $i \in [n_{\alpha}]$ the values $\{\beta_{i,j}\}_{j \in [n_{\beta}]}$ are distinct.
- 3. For every $i \in [n_{\alpha}]$ and $j \in [n_{\beta}]$ the values $\{\gamma_{i,j,k}\}_{k \in [n_{\gamma}]}$ are distinct.

Claim C.2. Let $q \in \mathbb{N}$ be a prime number, let $m, n \geq 1$, $a, b, c \in \mathbb{Z}_q^m$, $d \in \mathbb{Z}_q^n$, $e \in \mathbb{Z}_q$ and let $f : \mathbb{Z}_q^3 \to \mathbb{Z}_q$ be defined as

$$f(\alpha, \beta, \gamma) = \langle \boldsymbol{a}, \boldsymbol{\alpha}^m \circ \boldsymbol{\gamma}^m \rangle + \langle \boldsymbol{b}, \boldsymbol{\beta}^m \rangle + \langle \boldsymbol{c}, \boldsymbol{\gamma}^m \rangle + \langle \boldsymbol{d}, \boldsymbol{\alpha}^n \rangle + e.$$

If there exists a path-distinct set of $(n+1) \cdot (m+1)^2$ triplets $\{(\alpha_i, \beta_{i,j}, \gamma_{i,j,k})\}_{i \in [n+1], j,k \in [m+1]}$ such that $f(\alpha_i, \beta_{i,j}, \gamma_{i,j,k}) = 0$ for every $i \in [n+1]$ and $j,k \in [m+1]$, then $\mathbf{a} = \mathbf{b} = \mathbf{c} = 0^m$, $\mathbf{d} = 0^n$ and e = 0.

Proof. Fix any $i \in [n+1]$ and $j \in [m+1]$, and let

$$g_{i,j}(\gamma) = \langle \boldsymbol{a} \circ \boldsymbol{\alpha}_i^m + \boldsymbol{c}, \boldsymbol{\gamma}^m \rangle + \langle \boldsymbol{b}, \boldsymbol{\beta}_{i,j}^m \rangle + \langle \boldsymbol{d}, \boldsymbol{\alpha}_i^n \rangle + e$$
.

Then $g_{i,j}$ is a polynomial of degree m, and it holds that $g(\gamma_{i,j,k}) = 0$ for every $k \in [m+1]$. Therefore, $g_{i,j}$ is the zero polynomial, and thus its m+1 coefficients are all zeros. That is, $\langle \boldsymbol{b}, \boldsymbol{\beta}_{i,j}^m \rangle + \langle \boldsymbol{d}, \boldsymbol{\alpha}_i^n \rangle + e = 0$ and $\boldsymbol{a} \circ \boldsymbol{\alpha}_i^m + \boldsymbol{c} = 0^m$. For each $i \in [n+1]$, let

$$h_i(\beta) = \langle \boldsymbol{b}, \boldsymbol{\beta}^m \rangle + \langle \boldsymbol{d}, \boldsymbol{\alpha}_i^n \rangle + e$$
.

Then, h_i is a polynomial of degree m, and it holds that $h_i(\beta_{i,j}) = 0$ for every $j \in [m+1]$. Therefore, h_i is the zero polynomial, and thus its m+1 coefficients are all zeros. That is, $\boldsymbol{b} = 0^m$ and $\langle \boldsymbol{d}, \boldsymbol{\alpha}_i^n \rangle + e = 0$. Next, let

$$t(\alpha) = \langle \boldsymbol{d}, \boldsymbol{\alpha}^n \rangle + e$$
,

then t is a polynomial of degree n, and it holds that $g(\alpha_i) = 0$ for every $i \in [n+1]$. Therefore, t is the zero polynomial, and thus its n+1 coefficients are all zeros. That is, $\mathbf{d} = 0^n$ and e = 0. Finally, since for $n+1 \geq 2$ distinct values of α_i it holds that $\mathbf{a} \circ \boldsymbol{\alpha}_i^m + \mathbf{c} = 0^m$, then $\mathbf{a} = \mathbf{c} = 0^m$.