Proving e-voting mixnets in the CCSA model: zero-knowledge proofs and rewinding

Margot Catinaud, Caroline Fontaine, Guillaume Scerri

Proving e-voting mixnets in the CCSA model: zero-knowledge proofs and rewinding

Margot Catinaud[1], Caroline Fontaine[1], and Guillaume Scerri[1]

[1]Université Paris-Saclay, CNRS, ENS Paris-Saclay, Laboratoire Méthodes Formelles, 91190, Gif-sur-Yvette, France, firstname.lastname@lmf.cnrs.fr

February, 2025

Abstract

Mixnet protocols are used in electronic voting protocols to mix the ballot box before the tally, to preserve ballots privacy and unlinkabiliy. Whereas proving security properties of the other components of the electronic voting protocols has globally already been done in several logical frameworks and tools, proofs of mixnets remain a real challenge to handle. In this paper we focus on the quite recent CCSA logic, which enables handling of computational security proofs with first-order logics facilities. We enrich the logic to be able to deal with zero-knowledge proofs and rewinding techniques, and provide the first complete proof of Terelius-Wikström mixnet protocol.

1 Introduction

Electronic voting protocols are more and more used for widespread applications, from professional elections even to political elections, in direct democracy, as for example in Switzerland. Therefore, depending on the criticality of the elections, we need them to provide robust security guarantees. In broad terms, such protocols should achieve two main properties: verifiability and privacy. Roughly, verifiability ensures that all the ballots in the ballot box have indeed been counted during the tally (*universal verifiability*) and that each voter can verify if his ballot is present in the ballot box (*individual verifiability*). Besides, privacy hides a variety of shades, such as "no adversary can link each ballot to each voter". E-voting protocols are running in three main steps. They begin with a setup, where a server is prepared (by an authority commitee) to host the election. Next, it is strictly speaking the voting phase where voters submit their votes in a public *ballot box*. Finally, as soon as the voting delay expires, authorities bring together to perform the tally. However, some metadata artefacts may remain at this stage. For simple tally, for example when only sums are involved, homomorphic encryption can be enough to compute the result while eliminating these remainings artefacts. However, for more complex tallies, for example when voters are asked to rank candidates, we need to mix the ballot box to safely decrypt the ballots and then compute the tally without compromise ballot privacy, this thanks to mixnet protocols.

In a nutshell, a *mixnet* is composed of a certain number of authorities acting as a network. Each *mix-server* takes as input the encrypted list of ballots, and produces as output a permutation (going with a reencryption) of these ballots. Regarding the main security properties we want electronic voting protocols to guarantee, we expect two security properties from mixnets:

- *Permutation secrecy*: A honest mixnet should ensure that no adversary can link votes from the output list with votes from the input list.
- *Verifiability*: A dishonest mixnet should not be able to convince a honest verifier that the output list is a reencrypted permutation of the input list when it is not the case. In particular, as the output list is

indeed a permutation of the input list, no ballot can be dismissed or duplicated. This preserves both individual and universal verifiability properties.

To achieve these security properties, mixnets rely on advanced cryptographic constructions, such as zero-knowledge proofs and commitment schemes, with complex interactions between them. Consequently, proof techniques required by these protocols are quite evolved. Indeed, they are based on non-standard cryptographic reductions (e.g. rewinding) and complex interactions between cryptographic and algebraic results.

As a matter of fact, the complexity of the arguments needed to prove mixnets security may decrease our confidence regarding handmade proofs. As an example, the original pen-and-paper proof of the Terelius and Wikström [19, 18] mixnet (a variant of mixnet protocols used in electronic voting protocols CHVote [12] or Belenios [10]) provides key cryptographic and algebraic ideas but is far from a complete cryptographic proof. Providing a full and precise pen-and-paper cryptographic security proof based on this previous work would be a challenging and non-trivial task. Considering the size and complexity of the resulting proof, its formal verification will yield much higher confidence.

A number of tools aim at mechanizing security proofs of cryptographic models. These tools are based on two main paradigms: *symbolic models* where attackers consider cryptographic constructions as black-boxes with perfect secrecy, and *computational models* where attackers can break cryptography with some probability. Purely symbolic tools (such as ProVerif [7] or Tamarin [15]) do not fit to capture the fine grained probabilistic reasoning needed to faithfully catch all the key arguments of the proof. Therefore, this leaves with tools based on computational models, where attackers are modelled by probabilistic polynomial Turing machines. CryptoVerif [8] is a fully automated tool, which does not support arbitrary mathematical reasoning nor rewinding, which are both essential to perform security proofs of mixnets. Tools based on probabilistic Hoare logic (mainly EasyCrypt [6]) will be more suited for our purpose. Unfortunately, event if recent advances allow rewinding in EasyCrypt [11], performing complex proofs of protocols using advanced cryptographic techniques remains complex and time consuming with this kind of tools, making them ill suited for our goal.

In this work, we focus on a third paradigm, namely the *Computationally Complete Symbolic Attacker* (CCSA) model [4, 3]. This model aims to take benefits of both previous paradigms: automation of symbolic models to ease proofs thanks to a powerful logic, and strong cryptographic guarantees of computational models hidden in a probabilistic semantics. These benefits rely on the central predicate $u \sim v$ encoding the fact that the probability for a probabilistic polynomial time adversary to distinguish the computational interpretations of the terms u and v is negligible. In order to perform a proof in this model, one has to provide elementary axioms in this logic to capture the properties of the cryptographic constructions used (their computational interpretation should be proven sound) and then perform the proof with the help of those axioms. A correct proof then provides guarantees against a computational attacker. This logic has been implemented in the Squirrel proof assistant [2]. This logic allows relatively simple proofs of complex protocols (for example key-management APIs [16]), with very limited work on proving soundness of the axioms. Notably, the soundness proofs are small and relatively easy to check, and the remainder of the reasoning can be checked automatically.

1.1 Related work

The original works of Terelius and Wikström on mixnets [18, 19] provide a sketch of proof of the proposed constructions, with a particular focus on algebraic properties. Our proof structure follows their proof, which does not formalise the rewinding step, and simply assumes that the adversary produces as many proof transcript as needed to extract a number of witnesses. This formalisation is at the core of our contribution, and requires a number of non trivial reductions and probabilistic arguments that are not present in the original proof sketch. In particular, throughout the proof the Terelius and Wikström assumes that the adversary is not able to influence the distribution of challenges for which rewinding is performed; our formalisation allows us to identify the fact that this assumption does not hold and that all algebraic arguments must hold for vectors that are not necessarily uniformly random (see Section 5 for more details).

We have to mention that some efforts have already been made to formalise such proofs. In particular [13,

14] propose a model of Terelius-Wikström mixnet in Coq using the CertiCrypt project [5]. These proofs focus on properly capturing all arguments in the proof, except rewinding, at the associated probabilistic arguments. Their models are lower level when studying the algebraic properties (which we mostly axiomatise), providing a lot of confidence in the algebraic reasoning justifying in particular the proof of permutation. However, [13, 14] do not model rewinding at all and thus miss the adversarial selection argument exposed previously. This work is complementary to ours, it provides us with confidence that our axiomatisation of algebraic properties is correct, while our work ensures that the rewinding step of the proof is correct (once gaps are fixed).

Finally a number of works aim at formalising and automating cryptographic proofs. Most notably, CyptoVerif [8] and EasyCrypt [6] provide both a formal model for cryptographic proofs and some level of automation. CryptoVerif's semantics does not allow for capturing rewinding as the adversary is always implicit. For EasyCrypt logic's , some work has been performed for formalising rewinding [11]. However, proofs in EasyCrypt are notoriously involved as soon as the reductions become complex, and capturing the nested rewinding steps necessary here would be a rather complex problem in this logic. By contrast, we provide here a relatively simple proof thanks to our formalisation of adversarial success.

1.2 Contributions

Focusing on Belenios electronic voting protocol, many automated proofs have been proposed, in the symbolic [1] and in the computational [9] models. However, due to the complexity of mixnets security proofs, this very step of electronic voting protocols remains a blind spot in the security study of these works. Actually, the computational proofs of security focus on the cases where simple tally solved with homomorphic encryption is enough. As a matter of fact, proving mixnet security proof appears to be the last substantial barrier to obtain a fully automated security proof of the Belenios protocol against a computational attacker.

Our work aims to fulfil this gap with a proof of the Terelius-Wikström mixnet in the CCSA logic. This requires new axioms and an extension of the semantics of the logic. More precisely:

- We provide (and prove) axioms for the algebraic properties needed for the proof.
- We provide the first axiomatisation of zero-knowledge proofs, commitment schemes and reencryption.
- We provide a new construction that allows us to capture rewinding in the CCSA logic. Natively, the original CCSA logic only allows reasoning on globally negligible (or globally non negligible) events, in the sense that one can only reason on probabilities on the whole sampling space, but do not have any construction to deal conditional probabilities. However, rewinding requires reasoning on the probability of a certain event *knowing* that an execution point has been reached. Therefore, we introduce a new construction in the CCSA logic that captures that a certain formula is true with non negligible probability knowing that another formula is true. Additionally, we provide axioms related with the interaction between this construction and the usual global CCSA formulae. This allows us to capture the rewinding argument.

1.3 Outline

Our paper is organised as follows. We first provide some background on the CCSA logic in Section 2. Then, Section 3 is dedicated to the presentation of cryptographic primitives and the properties they address; it is followed by the presentation in Section 4 of Terelius-Wikström mixnet protocol. We provide the proofs of verifiability and permutation secrecy respectively in Sections 5 and 6. The article ends in Section 7 with a summary of contributions and future work directions.

2 The CCSA logic

We recall here the key concepts of the computationally complete symbolic attacker logic from [3]. This logic is built on (higher order) terms, using *names* to denote random samplings, and a subset of functions

for representing the adversarial comptations. These terms are interpreted as random variables over the randomness of both the protocol and the adversary, and represent the interactions of the protocol and the adversary.

Formulas are built on top of two main predicates: $[\phi]$ which denotes that a formula (a term of type **bool**) is true with overwhelming probability, and $\mathbf{u} \sim \mathbf{v}$ that states that no probabilistic polynomial time adversary can distinguish the distribution of the lists of terms $\mathbf{u}$ and $\mathbf{v}$ with non negligible probability.

2.1 Terms

Types in the CCSA logic are built on a set of *base types* $\mathbb{T}$ using the usual type arrow $\to$. Notably we assume that *base types* includes at least the types **bool**, **nat**, **real** and **msg** (models bitstrings).

A *type structure* $\mathbb{M}$ defines an interpretation $[\![\tau]\!]^\eta_{\mathbb{M}}$ for each base type $\tau \in \mathbb{T}$ and security parameter η. The interpretation of standard types is the standard one. Function types are defined as usual. A type is said to be finite if, for any η, its interpretation is finite.

The terms considered in the CCSA logic are simply-typed λ-terms built upon a set of variables $\mathcal{X}$:

$$t ::= x \mid t\ t \mid \lambda(x:\tau).t \mid \forall(x:\tau).t$$

where variables represent function arguments, logical variables and function symbols (e.g. cryptographic functions) declared in an environment.

An *environment* consists in variable declarations $(x : \tau)$ and variable definitions $(x : \tau = t)$. We assume that environments declare at least the standard Boolean operation (e.g. $\wedge, \vee$), integer operations, real operations, and a number of standard function (in particular an **if then** construct). Note that environments allow for well founded recursive definitions.

A *model* $\mathbb{M}$ for a term structure $\mathcal{E}$ consists, for every η, of two sets of random tapes $\mathbb{T}^h_{\mathbb{M},\eta}$ (the *honest* randomness) and $\mathbb{T}^a_{\mathbb{M},\eta}$ (the *adversarial* randomness). It defines, for every declared variable $(x : \tau)$, for every security parameter η a $[\![\tau]\!]^\eta_{\mathbb{M}}$ valued random variable $\rho \in \mathbb{T}^h_{\mathbb{M},\eta} \times \mathbb{T}^h_{\mathbb{M},\eta} \mapsto [\![x]\!]^{\eta,\rho}_{\mathbb{M}}$. This interpretation $[\![\cdot]\!]^{\eta,\rho}_{\mathbb{M}}$ is naturally lifted to terms. We require that usual functions are interpreted in the standard way.

Example 2.1. For all $n \in [\![1; N]\!]$, for all $i \in [\![1; n]\!]$, we define the term $\mathbf{i}$ to be the i-th canonical vector $\mathbf{u}_i \in \mathbb{Z}^n_{q_\eta}$ where, for all $j \in [\![1; n]\!]$, $(\mathbf{u}_i)_j = \delta_{ij}$. Therefore, we have $[\![\mathbf{i}]\!]^{\eta\,,\,\rho}_{\mathbb{M}:\mathcal{E}} = \mathbf{u}_i$ for all security parameter η and all random tape $\rho \in \Omega$.

We call a subset $\mathcal{N} \subset \mathcal{X}$ of variables *names*. These are used to denote honest random samplings. Names can only be declared in an environment and are of type $\tau_0 \to \tau_b$ where τ_b is a base type. Names are interpreted as a sequence of *independent identically distributed random samplings* from the honest randomness $\mathbb{T}^h_{\mathbb{M},\eta}$ to τ_b. In other terms we require that two different names, or the same name used with two different indices do not "use" the same part of the random tape. Contrary to [3] we do not require that τ_0 is a finite type, however we require that all formulas involving names are guarded by a condition ensuring that they only use for every η a finite number of indices, which achieves the same effect. This allows us to define a recursive term of type **nat** $\to$ **msg** that returns a list of randomness of arbitrary size. It is then only used under the assumption that its argument is bounded for every η.

2.2 Formulas

The formulas of the CCSA logic are standard first order formulas built on top of the first order terms defined previously, with predicates designed to capture cryptographic reasoning. We write $\tilde{\vee}, \tilde{\wedge}, \tilde{\exists}, \ldots$ for the usual global logical connector, in order to distinguish them from their local counterparts that appear in terms. The semantics of the logic is the usual first order semantics, where $\mathbb{M} \models F$ means that F holds in $\mathbb{M}$.

We now recall the definition of the main predicates of the CCSA logics from [3]. As we aim at capturing cryptographic properties, we need to define what is a small enough success probability for the adversary.

Definition 2.1. A function $f : \mathbb{N} \to \mathbb{R}$ is *negligible* if for all polynomials P, asymptotically $f(\eta) \leq \frac{1}{P(\eta)}$.

The predicate $[\phi]$ denote the fact the the formula ϕ (i.e. a term of type **bool**) is almost always true. Precisely $\mathbb{M} \models [\phi]$ if $\Pr_\rho \left[[\![\phi]\!]^{\rho,\eta}_{\mathbb{M}} \right]$ is negligible in η.

The predicate $\sim$ captures computational indistinguishability. If $\mathbf{u}$ and $\mathbf{v}$ are lists of terms with matching types, $\mathbf{u} \sim \mathbf{v}$ holds if for any probabilistic polynomial time Turing machine $\mathcal{A}$

$$\left| \Pr_\rho \left[\mathcal{A}(1^\eta, [\![\mathbf{u}]\!]^{\rho,\eta}_{\mathbb{M}}, \rho_a) \right] - \Pr_\rho \left[\mathcal{A}(1^\eta, [\![\mathbf{v}]\!]^{\rho,\eta}_{\mathbb{M}}, \rho_a) \right] \right|$$

is negligible in η. Note that $\mathcal{A}$ is given access to the adversarial randomness from the model.

The predicate $\mathbf{adv}(u)$ expresses that the term u can be computed by the adversary in polynomial time. The predicate **det** states that a term does not depend on randomness (i.e. is a constant for each η). The predicate $\mathbf{pbound}(u)$ states, for a term of type **nat**, that $[\![u]\!]^{\rho,\eta}_{\mathbb{M}}$ is bounded by a polynomial in η.

The logics from [3] is equiped with a proof system that allows to reason at two levels, the *local* level (i.e. for a fixed randomness), and *global* (i.e. first order reasoning on the predicates given above). A *global* judgement $\mathcal{E}; \Theta \vdash F$ states that F is entailed by the global hypotheses Θ in environment $\mathcal{E}$. Precisely

$$\models \mathcal{E}; \Theta \vdash F \text{ if } \models (\tilde{\wedge}\Theta) \tilde{\to} F.$$

A *local* judgement $\mathcal{E}; \Theta; \Gamma \vdash \phi$ states that under global hypotheses Θ, Γ almost always entails ϕ (a term of type **bool**). Precisely

$$\models \mathcal{E}; \Theta; \Gamma \vdash \phi \text{ if } \models (\tilde{\wedge}\Theta) \tilde{\to} [(\wedge\Gamma) \to \phi].$$

In order to ensure that terms never need to be evaluated on unbounded randomness, for every term k used as index for a names in ϕ or F, we have $\mathbf{pbound}(k)$ as a global hypothesis.

The proof system proposed in [3] provides with generic reasoning rules for logical connectives, together with a number of rules dealing with simple properties of the predicates which we do not recall here.

3 Modelling cryptographic properties

Now, let us present some cryptographic primitives we need to model Terelius & Wikström mixnet protocol. First, we model commitment schemes to reveal only a fingerprint of the permutation used. Next, we model Σ-protocols which are a kind of interactive zero-knowledge proofs used to prove the good behavior of a mixnet. Finally, we model an abstraction of the shuffle performed by a mix-server, with so-called *shuffle-friendly* maps.

3.1 Commitment schemes

Commitment schemes are used to commit to an information without revealing it directly; the commited information is first sealed and can eventually be revealed later, but its value cannot be modified between the commitment step and the opening step. We denote by $\mathcal{M}$ the set of messages we will commit to. More formally, a *commitment scheme for the set of messages* $\mathcal{M}$ is a pair $\mathbb{KS}(\mathcal{M}) = (\mathbf{gencomkey}, \mathbf{com})$ of algorithms where

- $[\![\mathbf{gencomkey}]\!]^{\eta;\rho}_{\mathbb{M};\mathcal{E}}$ defines an algorithm which outputs a commitment key ck and defines the set $\mathcal{R}^{\text{COM}}_{\mathcal{M}}$ of randoms used to commit, as well as the set $\mathcal{K}_{\mathcal{M}}$ of commit messages.
- $\mathbf{com} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg}$ is a deterministic algorithm outputting a commit message a; it takes as inputs a commitment key ck, a message $m \in \mathcal{M}$ and a randomness $r \in \mathcal{R}^{\text{COM}}_{\mathcal{M}}$.

A commitment scheme has two cryptographic properties: the *hiding* property and the *binding* property. In both properties, the commitment key ck is honestly computed by a setup oracle.

- **(hiding property)** The *hiding* property states that given a commit message a, no polynomial time adversary can break a to obtain an opening information (m, r) such that $a = \mathbf{com}\ ck\ m\ r$. The cryptographic game $\text{Hiding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta\,;\,\beta)$ used to formalize the hiding property is a classic left-right game with some secret bit $\beta \in \{0,1\}$. An adversary against this game is given by a pair of probabilistic polynomial time adversaries $\mathcal{A} = (\mathcal{A}_{setup}, \mathcal{A}_{guess})$ such that $\mathcal{A}_{setup}$ generates a challenge consisting of two messages $m_0, m_1 \in \mathcal{M}$ with $m_0 \neq m_1$, and $\mathcal{A}_{guess}$ tries to guess β from the output of the commitment oracle which has commited to the message m_β. The adversary $\mathcal{A}$ wins the game $\text{Hiding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta\,;\,\beta)$ when β is correctly guessed by $\mathcal{A}_{guess}$. Hence, we define the advantage of $\mathcal{A}$ against the hiding game to be

$$\forall\,\eta \in \mathbb{N}^*, \text{Adv}\left[\mathcal{A}, \text{Hiding}_{\mathbb{KS}(\mathcal{M})}(1^\eta\,;\,\beta)\right](\eta) \stackrel{\text{def}}{=} \left|\Pr_{\eta\in\mathbb{N}^*}\left[\ 0 \leftarrow \text{Hiding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta\,;\,\beta=0)\ \right] - \Pr_{\eta\in\mathbb{N}^*}\left[\ 1 \leftarrow \text{Hiding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta\,;\,\beta=1)\ \right]\right|$$

When this advantage is, at least, a non-negligible function in the security parameter η, the following rule is sound:

G.Com:Hide

$$\frac{\mathcal{E};\Theta;\varnothing \vdash_{\text{pptm}} \mathbf{u}, m_1, m_2 \qquad \mathcal{E};\Theta \vdash [\phi^{r,i}_{\text{rand}}(\mathbf{u}, m_1, m_2) \wedge \phi^{ck,n}_{\text{comkey}}(\mathbf{u}, m_1, m_2)]}{\mathcal{E};\Theta \vdash \mathbf{u}, \mathbf{com}\ (ck\ n)\ m_1\ (r\ i) \sim \mathbf{u}, \mathbf{com}\ (ck\ n)\ m_2\ (r\ i)}$$

- **(binding property)** The *binding* property states that a commit message a can only be opened to one message m, the one used to compute a. We can see this property as the collision resistance for hash functions where the function **com** is seen as a kind of hash. Therefore, the idea behind the cryptographic game $\text{Binding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta)$ is to leave the choice of challenge messages to the adversary $\mathcal{A}$. She has to produce two messages $m_1, m_2 \in \mathcal{M}$ with two randoms $r_1, r_2 \in \mathcal{R}^{\text{COM}}_{\mathcal{M}}$ and sends this two pairs (m_1, r_1) and (m_2, r_2) to the commit oracle producing honest commitments $c_1, c_2 \in \mathcal{K}_{\mathcal{M}}$ from these two pairs. The adversary $\mathcal{A}$ wins the game $\text{Binding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta)$ when $c_1 = c_2$ but $(m_1, r_1) \neq (m_2, r_2)$. Hence, we define the advantage of $\mathcal{A}$ against the binding game to be

$$\forall\,\eta \in \mathbb{N}^*, \text{Adv}\left[\mathcal{A}, \text{Binding}_{\mathbb{KS}(\mathcal{M})}(1^\eta)\right](\eta) \stackrel{\text{def}}{=} \Pr_{\eta\in\mathbb{N}^*}\left[\ 1 \leftarrow \text{Binding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}(1^\eta)\ \right]$$

When this advantage is, at least, a non-negligible function in the security parameter η, the following rule is sound:

L.Com:Bind

$$\frac{\mathcal{E};\Theta;\varnothing \vdash_{\text{pptm}} m_1, m_2, r_1, r_2 \qquad \mathcal{E};\Theta;\Gamma \vdash \phi^{ck,n}_{\text{comkey}}(m_1, m_2) \qquad \mathcal{E};\Theta;\Gamma \vdash \mathbf{com}\ (ck\ n)\ m_1\ r_1 = \mathbf{com}\ (ck\ n)\ m_2\ r_2}{\mathcal{E};\Theta;\Gamma \vdash m_1 = m_2}$$

3.2 Σ-protocols

Let $\mathcal{R} \subset \mathcal{X}_{\mathcal{R}} \times \mathcal{W}_{\mathcal{R}}$ be a polynomial-time computable binary relation. For pairs $(x, w) \in \mathcal{R}$, we denote by $x \in \mathcal{X}_{\mathcal{R}}$ the *statement*, and by $w \in \mathcal{W}_{\mathcal{R}}$ the *witness*. We define the set $\mathcal{L}_{\mathcal{R}} \stackrel{\text{def}}{=} \{x \mid \exists\, w \in \mathcal{W}_{\mathcal{R}}, (x, w) \in \mathcal{R}\}$ to be the language set of the binary relation $\mathcal{R}$. Besides, given a security parameter $\eta \in \mathbb{N}^*$ and a random tape $\rho \in \Omega$, the property $[\![\mathbf{zkp\text{-}rel}_{\mathcal{R}}\ x\ w]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}$ holds for a statement x and a witness w when $([\![x]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}, [\![w]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}) \in \mathcal{R}$. A *Σ-protocol for the binary relation* $\mathcal{R}$ is a 3-message protocol $\Sigma_{\mathcal{R}}$ between two agents, a prover $\mathcal{P}$ and a verifier $\mathcal{V}$. These agents are formalized as polynomial-time Turing machines, as well as the setup phase $\mathcal{S}$: $\Sigma_{\mathcal{R}} \stackrel{\text{def}}{=} (\mathcal{S}, \mathcal{P}, \mathcal{V})$. Note that the statement x is a public input of both $\mathcal{P}$ and $\mathcal{V}$ but the witness w is a private input of only $\mathcal{P}$. The prover $\mathcal{P}$ is split in two steps, the first one is a *commit message* which "setup"

the interaction. Then, after the prover $\mathcal{P}$ receives a *challenge* sends by the verifier $\mathcal{V}$, $\mathcal{P}$ responds to this challenge. A Σ-protocol $\Sigma_{\mathcal{R}}$ is then defined by three functions $\mathbf{zkp\text{-}com}_{\mathcal{R}}$, $\mathbf{zkp\text{-}res}_{\mathcal{R}}$ and $\mathbf{zkp\text{-}verif}_{\mathcal{R}}$ which corresponds to executions of honests prover $\mathcal{P}$ and verifier $\mathcal{V}$ such that

- $\mathbf{zkp\text{-}com}_{\mathcal{R}} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg}$ outputs a *commit message* α on input a witness-statement pair $(x, w) \in \mathcal{R}$;
- $\mathbf{zkp\text{-}res}_{\mathcal{R}} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg}$ outputs a *response message* $z(c)$ on input a witness-statement pair $(x, w) \in \mathcal{R}$ and a *challenge* c chosen *uniformly at random* in the challenge space and sends by the verifier $\mathcal{V}$ after he received the commit message α;
- $\mathbf{zkp\text{-}verif}_{\mathcal{R}} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{bool}$ takes as input a statement x and a *proof transcript* $\langle \alpha, c, z(c) \rangle$ and outputs a boolean $b : \mathbf{bool}$ whether or not the verifier is convinced by the proof transcript.

To ease notations, the function $\mathbf{zkp\text{-}prove}_{\mathcal{R}} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg}$ is a macro standing for

$$\mathbf{zkp\text{-}prove}_{\mathcal{R}}\ x\ w\ (r\ i) \stackrel{\text{def}}{=} \langle \mathbf{zkp\text{-}com}_{\mathcal{R}}\ x\ w, r\ i, \mathbf{zkp\text{-}res}_{\mathcal{R}}\ x\ w\ (r\ i) \rangle$$

We denote by $(\mathcal{P}(w) \rightleftharpoons_{\mathcal{R}}^{(\Sigma)} \mathcal{V})(x)$ *the* Σ*-protocol interaction* between the prover $\mathcal{P}$ and the verifier $\mathcal{V}$ as described above by the macro $\mathbf{zkp\text{-}prove}_{\mathcal{R}}$. Finally, functions $\mathbf{zkp\text{-}prove}_{\mathcal{R}}$ and $\mathbf{zkp\text{-}verif}_{\mathcal{R}}$ must verify the following equation

$$\mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ (\mathbf{zkp\text{-}prove}_{\mathcal{R}}\ x\ w\ (r\ i)) = \top.$$

Besides, a Σ-protocol must verify the two following properties

- **(special-soundness)** $\Sigma_{\mathcal{R}}$ is said to be *special-sound* when their exist of a polynomial-time extractor $\mathcal{E}_{\mathcal{R}}$ given by the function $\mathbf{zkp\text{-}extract}_{\mathcal{R}} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg}$ such that the witness extraction is possible when two proof transcripts $\mathbf{p}_{\mathcal{R}}^{(i)} \stackrel{\text{def}}{=} \langle \alpha, c_i, z(c_i) \rangle$ (where $i \in \{1, 2\}$) are accepted by the verifier for the same commit message α but for different challenges $c_1 \neq c_2$. Informally, $\Sigma_{\mathcal{R}}$ is special-sound when any prover producing a proof accepted by the verifier for the witness-statement pair $(x, w) \in \mathcal{R}$ then the prover "knows" the witness w. Therefore, the following rule is sound:

L.Σ-P:SpSound

$$\frac{\mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{i \in \{1,2\}} \mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ \langle \alpha, c_i, z(c_i) \rangle \qquad \mathcal{E}; \Theta; \Gamma \vdash c_1 \neq c_2}{\mathcal{E}; \Theta; \Gamma \vdash \mathbf{zkp\text{-}rel}_{\mathcal{R}}\ x\ (\mathbf{zkp\text{-}extract}_{\mathcal{R}}\ x\ \mathbf{p}_{\mathcal{R}}^{(1)}(c_1)\ \mathbf{p}_{\mathcal{R}}^{(2)}(c_2))}$$

  where $\mathbf{p}_{\mathcal{R}}^{(i)}(c_i) \stackrel{\text{def}}{=} \langle \alpha, c_i, z(c_i) \rangle$.

- **(honest-verifier zero-knowledge)** This property is surely the trickiest one. The idea behind this property is to state that any proof accepted by the honest verifier $\mathcal{V}$ leaks no information about a witness w of a witness-statement pair $(x, w) \in \mathcal{R}$. More precisely, $\Sigma_{\mathcal{R}}$ is said to be *honest-verifier zero-knowledge* (HVZK) when there exists a polynomial-time simulator $\mathcal{S}im_{\mathcal{R}}$ (given by the function $\mathbf{zkp\text{-}sim}_{\mathcal{R}} : \mathbf{msg} \to \mathbf{msg} \to \mathbf{msg}$) such that on a statement $x \in \mathcal{L}_{\mathcal{R}}$ and a random challenge c outputs an accepting interaction $\langle \alpha, c, z \rangle$ with the same probability distribution as honest interactions $(\mathcal{P}(w) \rightleftharpoons_{\mathcal{R}}^{(\Sigma)} \mathcal{V}(c))(x)$ between the honest prover $\mathcal{P}$ and the honest verifier $\mathcal{V}$ where w is the witness for the statement x (*i.e.* $(x, w) \in \mathcal{R}$) and the verifier $\mathcal{V}$ must sends the challenge c. Hence, the following rule is sound.

G.Σ-P:HVZK

$$\frac{\mathcal{E}; \Theta; \varnothing \vdash_{\text{pptm}} \mathbf{u}, x, w \qquad \mathcal{E}; \Theta \vdash [\phi_{\text{rand}}^{r,i}(\mathbf{u}, x, w)]}{\mathcal{E}; \Theta \vdash \mathbf{u}, \mathbf{zkp\text{-}prove}_{\mathcal{R}}\ x\ w\ (r\ i) \sim \mathbf{u}, \mathbf{zkp\text{-}sim}_{\mathcal{R}}\ x\ (r\ i)}$$

In the case of Terelius & Wikström protocol, we will define a family of Σ-protocols $(\Sigma_{\mathcal{R}}(\mathbf{e}))_{\mathbf{e} \in \mathbb{Z}_{q_\eta}^N}$ for a family of binary relation $(\mathcal{R}(\mathbf{e}))_{\mathbf{e} \in \mathbb{Z}_{q_\eta}^N}$ for each vector $\mathbf{e} \in \mathbb{Z}_{q_\eta}^N$. Such family of Σ-protocols is defined by adding a first challenge vector $\mathbf{e} \in \mathbb{Z}_{q_\eta}^N$ sends by the verifier at the very beginning of the Σ-protocol. Next, the rest of the protocol is a standard Σ-protocol but for a relation depending of the challenge vector $\mathbf{e}$.

3.3 *Shuffle-friendly* maps

In their works, Terelius & Wikström generalize the rerandomization of the encryption and potential partial decryption performed by a mix-server, by a so-called *shuffle-friendly* map ϕ_{pk}. Formally, each ballot in the ballot box is encrypted using a cryptosystem $\mathbb{CS}$ allowing re-encryption (typically, a homomorphic cryptosystem is well-suited to encrypt ballots). The encryption algorithm $\text{Enc}_{\mathbb{CS}}$ is a non-deterministic algorithm using some randoms r as randomness: for a plaintext $m \in \mathcal{M}_{\mathbb{CS}}$ the encryption of m under the public key $pk = \texttt{pk}_{\mathbb{CS}}(sk)$ is $c = \text{Enc}_{pk}(m\,;\,r)$, where $r \xleftarrow{\$} \mathcal{R}_{\mathbb{CS}}$ is chosen uniformly at random. With this ciphertext c, we can re-encrypt the plaintext m without decrypting the ciphertext, by multiplying c by the encryption of 1 using some other random $r' \xleftarrow{\$} \mathcal{R}_{\mathbb{CS}}$. That is, if c' is the new ciphertext, then the re-encryption algorithm $\text{ReEnc}_{\mathbb{CS}}$ applies the following operation: $c' = \text{ReEnc}_{pk}(c\,;\,r') \stackrel{\text{def}}{=} c \cdot \text{Enc}_{pk}(1\,;\,r')$. Said differently, in the case of homomorphic cryptosystems, if $c = \text{Enc}_{pk}(m\,;\,r)$ and $c' = \text{ReEnc}_{pk}(c\,;\,r')$ then $c' = \text{Enc}_{pk}(m\,;\,r+r')$. A ciphertext c is said to be *well-formed for a secret key sk*, denoted by **wf_ctxt** $sk\ c$, when c can be decrypted with the secret key sk. Said otherwise,

$$[\![\textbf{wf\_ctxt}\ sk\ c]\!]^{\eta\,;\,\rho}_{\mathbb{M}\,:\,\mathcal{E}} = 1 \iff \text{Dec}_{[\![sk]\!]^{\eta\,;\,\rho}_{\mathbb{M}\,:\,\mathcal{E}}}([\![c]\!]^{\eta\,;\,\rho}_{\mathbb{M}\,:\,\mathcal{E}}) \neq \perp$$

We extend this predicate to ciphertexts lists $\mathbf{c}$ of length n:

$$\textbf{wf\_ctxt}_n\ sk\ \mathbf{c} \leftrightarrow \bigwedge_{i=1}^{n} (\textbf{wf\_ctxt}\ sk\ \langle \mathbf{c} \mid \mathbf{i} \rangle)$$

For a public key $pk \in \mathcal{PK}_{\mathbb{CS}}$, a map $\phi_{pk} : \mathcal{C}_{\mathbb{CS}} \times \mathcal{R}_{\mathbb{CS}} \longrightarrow \mathcal{C}_{\mathbb{CS}}$ is called *a shuffle-friendly map for a cryptosystem* $\mathbb{CS}$ if it defines a homomorphic map, *i.e.*, for all ciphertexts $c, c' \in (\mathcal{C}_{\mathbb{CS}}, \cdot)$ using some public key $pk \in \mathcal{PK}_{\mathbb{CS}}$ and for all randomness $r, r' \in (\mathcal{R}_{\mathbb{CS}}, +)$, we have $\phi_{pk}(c \cdot c'\,;\,r + r') = \phi_{pk}(c\,;\,r) \cdot \phi_{pk}(c'\,;\,r')$. We model these *shuffle-friendly* maps ϕ_{pk} in the CCSA model by supply a function **shuf-map** : **msg** $\rightarrow$ **msg** $\rightarrow$ **msg** $\rightarrow$ **msg**. Roughly, two different modes can be considered, separately or together: re-encryption or partial decryption. For the "*re-encryption only*" mode, the shuffle-friendly map ϕ^{renc}_{pk} is defined by the following equation:

$$\forall\, c \in \mathcal{C}_{\mathbb{CS}}, \forall\, r' \in \mathcal{R}_{\mathbb{CS}}, \phi^{\text{renc}}_{pk}(c\,;\,r) \stackrel{\text{def}}{=} \text{ReEnc}_{pk}(c\,;\,r').$$

Concerning partial decryption, we suppose we use a threshold cryptosytem where each mix-server $\mathbb{M}_j$, with $j \in [\![1; p]\!]$, of a mixnet $\mathbb{M} \stackrel{\text{def}}{=} (\mathbb{M}_j)_{j=1}^{p}$ has its own partial secret key sk_j. Moreover, we suppose that the public key pk of the election is computed from those partial secret keys $(sk_j)_{j=1}^{p}$, *i.e.* $pk = \texttt{pk}_{\mathbb{CS}}(sk_1, \ldots, sk_p)$, where at least $t \in [\![1; p]\!]$ of them are needed to be able to decrypt the whole election. Therefore, in the "*partial decryption only*" mode, the shuffle-friendly map ϕ^{dp}_{pk} is defined for each mix-server $\mathbb{M}_j$ by the following equation

$$\forall\, c \in \mathcal{C}_{\mathbb{CS}}, \forall\, r' \in \mathcal{R}_{\mathbb{CS}}, \phi^{\text{dp}}_{\texttt{pk}_{\mathbb{CS}}(sk_j)}(c\,;\,r') \stackrel{\text{def}}{=} \text{Dec}_{sk_j}(c).$$

In the CCSA logic, we denote by $\textbf{dec}_{\mathbb{CS}}$ the decryption predicate of only one ciphertext and by $\textbf{dec-list}^{(n)}_{\mathbb{CS}}$ the decryption of a ciphertext list of length n.

Actually, to be used in a mixnet protocol, a *shuffle-friendly* map ϕ_{pk} must verifies three properties.

- **(decryption preservation)** Firstly, the map ϕ_{pk} must keep untouched the content of each ballot. For this property, we suppose that the public key pk is honestly computed from a secret key sk, *i.e.* we have $pk = \texttt{pk}_{\mathbb{CS}}\ sk$. Therefore, we say that ϕ_{pk} *preserves decryption* when, for all ciphertexts $c, c' \in \mathcal{C}_{\mathbb{CS}}$ such that if (i) c is an encryption of a message $m \in \mathcal{M}_{\mathbb{CS}}$ using the public key pk, (*i.e.* we have $\exists\, r \in \mathcal{R}_{\mathbb{CS}}, \exists\, m \in \mathcal{M}_{\mathbb{CS}}, c = \text{Enc}_{pk}(m\,;\,r)$) and (ii) their exists a random value $r' \in \mathcal{R}_{\mathbb{CS}}$ such that $c' = \phi_{pk}(c\,;\,r')$ then we conclude $\text{Dec}_{sk}(c') = \text{Dec}_{sk}(c) = m$. Hence, when ϕ_{pk} preserves decryption, the following rule is sound.

L.Sfm:Correct

$$\frac{\mathcal{E}; \Theta; \Gamma \vdash \textbf{wf\_ctxt}\ sk\ c \qquad \mathcal{E}; \Theta; \Gamma \vdash \exists\, v.\ c' = \textbf{shuf-map}\ (\texttt{pk}_{\mathbb{CS}}\ sk)\ c\ v}{\mathcal{E}; \Theta; \Gamma \vdash \textbf{dec}_{\mathbb{CS}}\ sk\ c = \textbf{dec}_{\mathbb{CS}}\ sk\ c'}$$

Notice that for each new definition of a *shuffle-friendly* maps, one has to prove this new map verifies the decryption preservation property.

- **(associated zero-knowledge proof)** Secondly, we suppose we can prove with a zero-knowledge proof that a ciphertext $c' \in \mathcal{C}_{\mathbb{CS}}$ is computed with ϕ_{pk} from a ciphertext $c \in \mathcal{C}_{\mathbb{CS}}$ and a random value $r' \in \mathcal{R}_{\mathbb{CS}}$. Hence, we define the following computable binary relation for *shuffle-friendly* map $\mathcal{R}^{\text{map}}_{\phi_{pk}}$ to be the following relation

$$((pk, c, c'), r') \in \mathcal{R}^{\text{map}}_{\phi_{pk}} \overset{\text{def}}{\iff} c' = \phi_{pk}(c\,;\, r').$$

- **(indistinguishability of ϕ_{pk} output)** Finally, we do not want that the map ϕ_{pk} leak any information about the inputted ciphertext $c \in \mathcal{C}_{\mathbb{CS}}$. Let $c, c' \in \mathcal{C}_{\mathbb{CS}}$ be two ciphertexts such that $c' = \phi_{pk}(c\,;\, r')$ and $c = \text{Enc}_{pk}(m\,;\, r)$ where $m \in \mathcal{M}_{\mathbb{CS}}$ is a message and $r, r' \in \mathcal{R}_{\mathbb{CS}}$ are random values. In the case of "*re-encryption only*" mode, a solution will be to suppose that no adversary can distinguish between a re-encrypted ciphertext $c' = \text{ReEnc}_{pk}(c\,;\, r')$ from a *first-time* encryption, *i.e.* in the case where $c' = \text{Enc}_{pk}(m\,;\, r')$. However, we consider this solution to be too strong for the use we want. Indeed, we only want that any adversary can not distinguish if c' was computed by ϕ_{pk} using a ciphertext $c_1 \in \mathcal{C}_{\mathbb{CS}}$ or some other $c_2 \in \mathcal{C}_{\mathbb{CS}}$. Besides, this last solution seems to be more suitable for the "*partial decryption only*" mode. Therefore, we define a new cryptographic game $\text{Ind-CPA}^{\mathcal{A}}_{\phi_{pk},\,\mathbf{valid}}(\eta\,;\,\beta)$ to be a left-right game with some secret $\beta \in \{0, 1\}$. An adversary against this game is given by a pair of probabilistic polynomial time adversaries $\mathcal{A} = (\mathcal{A}_{setup}, \mathcal{A}_{guess})$. The first sub-adversary $\mathcal{A}_{setup}$ generates two ciphertexts $c_0, c_1 \in \mathcal{C}_{\mathbb{CS}}$ with $c_0 \neq c_1$ and a proof v that these ciphertexts verify any valid predicate **valid** which is an over-approximation property of the well-formed predicate **wf_ctxt**:

$$\mathbf{valid}\ pk\ c\ v \to \exists\ sk.\ pk = \mathtt{pk}_{\mathbb{CS}}\ sk\ \wedge\ \mathbf{wf\_ctxt}\ sk\ c.$$

Then, we extend this predicate to ciphertexts list $\mathbf{c}$ in this way

$$\mathbf{valid}_n\ pk\ \mathbf{c}\ v \to \exists\ sk.\ pk = \mathtt{pk}_{\mathbb{CS}}\ sk\ \wedge\ \mathbf{wf\_ctxt}_n\ sk\ \mathbf{c} \wedge \bigwedge_{1 \leqslant i < j \leqslant n} (\mathbf{len}\ \langle \mathbf{c} \mid \mathbf{i} \rangle = \mathbf{len}\ \langle \mathbf{c} \mid \mathbf{j} \rangle)$$

Next, the second sub-adversary $\mathcal{A}_{guess}$ tries to guess β from the output of the oracle which apply the map ϕ_{pk} to the ciphertext c_β. Therefore, the adversary $\mathcal{A}$ wins the game $\text{Ind-CPA}^{\mathcal{A}}_{\phi_{pk},\,\mathbf{valid}}(\eta\,;\,\beta)$ when β is correctly guessed by $\mathcal{A}_{guess}$. When $\mathcal{A}$ wins the game with negligible advantage, the following rule is sound.

G.Sfm:IndCPA

$$\frac{\mathcal{E}; \Theta; \varnothing \vdash_{\text{pptm}} \mathbf{u}, c, v \qquad \mathcal{E}; \Theta \vdash [\phi^{sk}_{\text{skey}}(\mathbf{u}, c, v)]}{\mathcal{E}; \Theta \vdash \begin{array}{l} \mathbf{u}, \mathbf{if\ valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ v\ \mathbf{then\ shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ r \\ \sim\ \mathbf{u}, \mathbf{if\ valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ v \\ \quad \mathbf{then\ shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{0}\ (\mathbf{len}\ c))\ r \end{array}}$$

Example 3.1. Using the El-Gamal cryptosystem over a group $\mathbb{G}_{q_\eta}$ with public key $pk = (g, y)$, where $y = g^{sk}$ with secret key sk, we define the *shuffle-friendly* map defined by $\phi_{(g,y)}((u, v)\,;\, r) = (g^r u, y^r v)$ describes re-encryption. Besides, if $y_i = g^{sk_i}$ is a partial key, $y = y_1 y_2 y_3$ is the shared public key, and $sk = sk_1 + sk_2 + sk_3$ is the shared secret key, then the *shuffle-friendly* map defined by $\phi^{sk_1}_{(g,y)}((u, v)\,;\, r) = (g^r u, (y/y_1)^r u^{-sk_1} v)$ denotes partial decryption and re-encryption using the partial secret sk_1 and randomness r. Both *shuffle-friendly* maps defined above satisfy security properties given above. For the map $\phi_{(g,y)}$ for re-encryption, the indistinguishability of $\phi_{(g,y)}$ outputs property holds because the El-Gamal cryptosystem satisfies the Ind-CPA assumption. In the case of $\phi^{sk_1}_{(g,y)}$ for partial decryption and re-encryption, the indistinguishability property holds because of the *Decisional Diffie-Hellman* assumption.

4 Terelius and Wikström mixnet protocol

Before presenting the protocol, we need to introduce some notation. From now on, N will denote a non-null natural number and $q_\eta \in \mathbb{N}^*$ will be a prime number of size at least η, *i.e.* we have $\log_2 q_\eta \geqslant \eta$. Moreover, $\mathbb{G}_{q_\eta}$ refers to a cyclic group of order q_η. We denote by $\langle \cdot \mid \cdot \rangle$ the standard scalar product over $\mathbb{Z}_{q_\eta}^N$: for all vectors $\mathbf{x} = (x_1, \ldots, x_N), \mathbf{y} = (y_1, \ldots, y_N) \in \mathbb{Z}_{q_\eta}^N$, we have $\langle \mathbf{x} \mid \mathbf{y} \rangle = \sum_{i=1}^N x_i y_i$. By $\mathbf{1}$, we refer to the unit vector $\mathbf{1} = (1, \ldots, 1) \in \mathbb{Z}_{q_\eta}^N$. Finally, we define $\circledast$ to be the following operator on vectors: for two vectors $\mathbf{x} = (x_1, \ldots, x_N), \mathbf{y} = (y_1, \ldots, y_N) \in \mathbb{Z}_{q_\eta}^N$, $\mathbf{x} \circledast \mathbf{y} = \prod_{i=1}^N x_i^{y_i}$. In Terelius-Wikström mixnet protocol, we choose to represent permutations as matrices. More precisely, if $\pi \in \mathfrak{S}_N$ is a permutation of length N, the representation of the permutation π in the form of matrix is given by the matrix $M_\pi = \left(m_{i,j}^{(\pi)}\right)_{1 \leqslant i,j \leqslant N}$ where, for all $i, j \in [\![1; N]\!]$, $m_{i,j}^{(\pi)} = \delta_{i\pi(j)}$. We denote by $\mathbf{perm}_N\ M_\pi$ when M_π is a permutation matrix.

4.1 Protocol description

Before presenting the key ideas of the protocol, let us recall informally the properties we want a mix-server to satisfy:

- **(Correctness)** When both mix-server and verifier are honest, a mix-server must keep the content of each ballot untouched, and the proof transcripts produced by the mix-server must be accepted by the verifier. More precisely, the decryption of the inputted list of ballots and the decryption of the outputted list of ballots are equal in the sense of multisets. Actually, as this property has been showned in [18] and [14], we won't linger on it.
- **(Permutation secrecy)** When the mix-server is honest but the verifier is dishonest, *i.e.* controlled by an adversary $\mathcal{A}$, the mix-server blurs the link between the outputted list of ballots and the inputted one. That is, the adversary $\mathcal{A}$ cannot link each ballot to each voter.
- **(Verifiability)** This property aims to verify that a mix-server does not cheat, under the assumptions that the mix-server is controlled by an adversary $\mathcal{A}$ and the verifier is honest. More precisely, it is achieved if the adversary $\mathcal{A}$ cannot not produce proofs transcripts accepted by the verifier while the decryption of the outputted list of ballots is not a permutation of the inputted one.

The mixnet protocol proposed by Terelius and Wikström ([19], [18]) is split into two parts, one *offline* and one *online*. We will describe it briefly, before diving deeper into the details.

The protocol in a nutshell: First, at the same time as the election setup, and during the *offline phase*, each mix-server chooses a random permutation $\pi \xleftarrow{\$} \mathfrak{S}_N$ and publishes a commitment to the matrix M_π representing this permutation π. In other words, each mix-server chooses a random vector $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N$ and publishes the value $\mathbf{a} \leftarrow \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N \times N}}(ck, M_\pi\,; \mathbf{s})$ where the commitment algorithm $\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N \times N}}$ is based on Pedersen's commitment scheme. In doing so, each mix-server publicly promises to use the permutation π without revealing itself. Later, just before the tally of the election, the *online phase* will consist of the ballot box mixing procedure, which goal is to blur the link between the lists of ballots and voters, ensuring ballot privacy. During this phase, each mix-server takes on its turn all the ballots in the ballot box $\mathbf{b}^{(\mathrm{in})}$ and outputs a permutted and re-randomized version of these ballots $\mathbf{b}^{(\mathrm{out})}$. Besides, each phase of the protocol comes with a zero-knowledge proof attesting that the target property is satisfied.

More details: To define a commitment scheme for matrix, we first define a commitment scheme for vectors in $\mathbb{Z}_{q_\eta}^N$ based on Pedersen's commitment scheme. The commitment algorithm **com-vec** for vectors in $\mathbb{Z}_{q_\eta}^N$ is given by the following function.

$$\begin{array}{rccc} \mathrm{Com}_{\mathbb{Z}_{q_\eta}^N} : & \mathbb{G}_{q_\eta}^{N+1} \times \mathbb{Z}_{q_\eta}^N \times \mathbb{Z}_{q_\eta} & \longrightarrow & \mathbb{G}_{q_\eta} \\ & \big((g, \mathbf{g}), \mathbf{x}, s\big) & \longmapsto & g^s \prod_{i=1}^N g_i^{x_i}. \end{array}$$

Next, the commitment algorithm **com-mat** for matrix in $\mathbb{Z}_{q_\eta}^{N\times N}$ is based on this commitment scheme for vectors **com-vec** with the exception that the randomness space is $\mathbb{Z}_{q_\eta}^{N}$ and the commitment space is $\mathbb{G}_{q_\eta}^{N}$. For a matrix M, a commitment key ck and a random vector $\mathbf{s}$, the commit message $\mathbf{a} = \textbf{com-mat}\ ck\ M\ \mathbf{s}$ to the matrix M is such that $\langle \mathbf{a} \mid \mathbf{i}\rangle \stackrel{\text{def}}{=} \textbf{com-vec}\ ck\ (M\cdot\mathbf{i})\ \langle \mathbf{s} \mid \mathbf{i}\rangle$. Both previous commitment schemes are *perfectly hiding* and *computationally binding* under the *Discrete Logarithm* assumption for the group $\mathbb{G}_{q_\eta}$. During the *offline* phase, each mix-server must produce a valid commit message to the secret permutation matrix it chose. The corresponding zero-knowledge proof proving this step is based on an algebraic result of characterisation of permutation matrix. Indeed, a matrix $M \in \mathbb{Z}_{q_\eta}^{N\times N}$ is a permutation matrix as soon as (i) $M\cdot\mathbf{1} = \mathbf{1}$ and (ii) for all vector $\mathbf{e} = (e_1,\ldots,e_N) \in \mathbb{Z}_{q_\eta}^{N}$, $\prod_{i=1}^{N}(M\cdot\mathbf{e})_i = \prod_{i=1}^{N} e_i$.

Definition 4.1 (Correct commitment relation)**.** Let $\mathbf{a} \in \mathbb{G}_{q_\eta}^{N}$ be a vector. Let $ck \leftarrow \text{Gen}_{\mathbb{Z}_{q_\eta}^{N}}(1^\eta, N) \in \mathbb{G}_{q_\eta}^{N+1}$ be a commitment key of the Pedersen commitment scheme $\mathbb{KS}(\mathbb{Z}_{q_\eta}^{N})$. Let $\mathbf{e} \in \mathbb{Z}_{q_\eta}^{N}$ be a vector. We define $\mathcal{R}^{\text{com}}(\mathbf{e})$ to be the *(binary) relation of correct commitment for the vector* $\mathbf{e}$ where

$$((ck,\mathbf{a}),(t,\mathbf{e}',k)) \in \mathcal{R}^{\text{com}}(\mathbf{e}) \stackrel{\text{def}}{\iff} \begin{cases} \mathbf{a}\circledast\mathbf{1} = \text{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck,\mathbf{1}\,;\,t) \\ \wedge\ \mathbf{a}\circledast\mathbf{e} = \text{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck,\mathbf{e}'\,;\,k) \\ \wedge\ \prod_{i=1}^{N} e'_i = \prod_{i=1}^{N} e_i. \end{cases}$$

Roughly, if $\mathbf{a} = \text{Com}_{\mathbb{Z}_{q_\eta}^{N\times N}}(ck, M\,;\,\mathbf{s})$ for some matrix $M \in \mathbb{Z}_{q_\eta}^{N\times N}$ and some vector $\mathbf{s} \in \mathbb{Z}_{q_\eta}^{N}$, by operation on commitments, we obtain $\text{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck, M\cdot\mathbf{1}\,;\,\langle\mathbf{s}\mid\mathbf{1}\rangle) = \text{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck,\mathbf{1}\,;\,t)$. Therefore, by applying the binding property for the commitment scheme on vectors, we obtain the first condition of the characterisation of permutation matrix, *i.e.* (i) $M\cdot\mathbf{1} = \mathbf{1}$. By doing the same thing, we obtain the second part of the relation of correct commitments $M\cdot\mathbf{e} = \mathbf{e}'$, and then (ii) $\prod_{i=1}^{N}(M\cdot\mathbf{e})_i = \prod_{i=1}^{N} e_i$. Therefore, if $\mathbf{a}$ is a commit message to a matrix M using a commitment key ck and if the relation of correct commitment $\mathcal{R}^{\text{com}}(\mathbf{e})$ holds for some vector $\mathbf{e} \in \mathbb{Z}_{q_\eta}^{N}$ computed by the verifier and for the statement $(ck,\mathbf{a})$, *i.e.* $(ck,\mathbf{a}) \in \mathcal{L}_{\mathcal{R}^{\text{com}}(\mathbf{e})}$, then M is a permutation matrix. During the *online* phase, a mix-server has to use the same permutation (than the one picked up and commited during the *offline* phase) to permut the inputted list of ballots and transforms it thanks to the *shuffle-friendly* map ϕ_{pk}. We define the following relation of correct shuffle.

Definition 4.2 (Correct shuffle relation)**.** Let $\mathbf{a} \in \mathbb{G}_{q_\eta}^{N}$ be a vector of size N. Let $ck \leftarrow \text{Gen}_{\mathbb{Z}_{q_\eta}^{N\times N}}(1^\eta, N) \in \mathbb{G}_{q_\eta}^{N+1}$ be a commitment key for the Pedersen commitment scheme $\mathbb{KS}(\mathbb{Z}_{q_\eta}^{N\times N})$. Let $(sk,pk) \leftarrow \text{KeyGen}_{\mathbb{CS}}(1^\eta) \in \mathbb{Z}_{q_\eta}\times\mathcal{PK}_{\mathbb{CS}}$ be a key pair for the cryptosystem $\mathbb{CS}$. Let ϕ_{pk} be a *shuffle-friendly* map for the cryptosystem $\mathbb{CS}$. Let $\mathbf{c},\mathbf{c}' \in \mathcal{C}_{\mathbb{CS}}^{N}$ be two lists of ciphertexts. Let $\mathbf{e} \in \mathbb{Z}_{q_\eta}^{N}$ be a vector. We define $\mathcal{R}_{\phi_{pk}}^{\text{shuffle}}(\mathbf{e})$ to be *the (binary) relation of correct shuffle for the vector* $\mathbf{e}$ where

$$((ck,\mathbf{a},pk,\mathbf{c},\mathbf{c}'),(\mathbf{e}',k,u)) \in \mathcal{R}_{\phi_{pk}}^{\text{shuffle}}(\mathbf{e}) \stackrel{\text{def}}{\iff} \begin{cases} \mathbf{a}\circledast\mathbf{e} = \text{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck,\mathbf{e}'\,;\,k) \\ \wedge\ ((pk,\mathbf{c}\circledast\mathbf{e},\mathbf{c}'\circledast\mathbf{e}'),u) \in \mathcal{R}_{\phi_{pk}}^{\text{map}}. \end{cases}$$

Again, if $\mathbf{a}$ is a commit message to a matrix M, we obtain $M\cdot\mathbf{e} = \mathbf{e}'$ from the first equality and by the binding property for the commitment scheme to vectors. Hence, by an algebraic argument we will explain later, we conclude with overwhelming probability the existency of a vector of random values $\mathbf{r} = (r_i)_{i=1}^{N} \in \mathbb{Z}_{q_\eta}^{N}$ such that we have, for all $i \in [\![1;N]\!]$, $c'_{\pi(i)} = \phi_{pk}(c_i\,;\,r_i)$. To ease notations, we denote by $\textbf{shuffle}\ pk\ \mathbf{c}\ \pi\ \mathbf{r}$ the function outputting the ciphertext list $\mathbf{c}$. Finally, concrete definitions of Σ-protocols for both relations of correct commitment $\mathcal{R}^{\text{com}}(\mathbf{e})$ and of correct shuffle $\mathcal{R}_{\phi_{pk}}^{\text{shuffle}}(\mathbf{e})$ can be found in [18].

4.2 Algebraic properties

Proofs of verifiability strongly rely on some algebraic properties. Firstly, once enough witnesses are extracted and give enough equations to fully determine a matrix M and a vector $\mathbf{s}$ such that $\mathbf{a} = \textbf{com-vec}\ ck\ M\ \mathbf{s}$ (if M is of size N, we need N equations and therefore N witnesses), their exists a function $\textbf{solve} : \textbf{msg} \rightarrow$

$\mathbf{msg} \to \mathbf{msg} \to (\mathbf{msg} \times \mathbf{msg})$ whose semantics corresponds to an adaptation of the Gaussian elimination, which is a polynomial-time algorithm. Each witness $(t, \mathbf{e}'_i, k_i) \in \mathcal{W}_{\mathcal{R}}$ for the relation of correct commitment $\mathcal{R}^{\mathrm{com}}(\mathbf{e}_i)$ associated with the vector $\mathbf{e}_i \in \mathbb{Z}_{q_\eta}^N$ for all $i \in [\![1; N]\!]$ gives the following equation on the matrix M

$$\mathbf{a} \circledast \mathbf{e}_i = \mathrm{Com}_{\mathbb{Z}_{q_\eta}^N}(ck, \mathbf{e}'_i\,;\, k_i).$$

Actually, we have enough equations, *i.e.* we have N equations, when the vector family $(\mathbf{e}_i)_{i=1}^N$ defines a basis of the vector space $\mathbb{Z}_{q_\eta}^N$. In that case, we denote $\mathbf{basis}_N\ (\mathbf{e}_i)_{i=1}^N$ in the CCSA logic. As $\dim(\mathbb{Z}_{q_\eta}^N) = N$, we only need a free family, which is achieved with overwhelming probability for any vector family chosen uniformly and independently at random. Therefore, we can model the opening of the commit value $\mathbf{a}$ by the following rule:

L.Open

$$\frac{\mathcal{E}; \Theta; \Gamma \vdash \mathbf{basis}_N\ (\mathbf{e}_i)_{i=1}^N \qquad \mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{i=1}^N (\mathbf{a} \circledast \mathbf{e}_i = \mathbf{com\text{-}vec}\ ck\ \mathbf{e}'_i\ k_i)}{\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ ck\ M\ \mathbf{s}}$$

where $(M, \mathbf{s}) \overset{\text{def}}{=} \mathbf{solve}\ \mathbf{a}\ (\mathbf{e}_i)_{i=1}^N\ (\mathbf{e}'_i, k_i)_{i=1}^N$.

Secondly, when the matrix M is obtained, we use the characterisation of permutation matrix to show that this matrix indeed represents a permutation. This characterisation states that M is a permutation matrix *if and only if* the two following equations hold: (i) $M \cdot \mathbf{1} = \mathbf{1}$ and (ii) when $\mathbf{e}$ is chosen uniformly at random in $\mathbb{Z}_{q_\eta}^N$, then $\prod_{i=1}^N (M \cdot \mathbf{e})_i = \prod_{i=1}^N e_i$. In the CCSA model, we denote this last product operation by the function $\mathbf{prod}_N$. Actually, to model this characterisation result in the CCSA model, the second condition (ii) is a bit twisted. Indeed, instead of Condition (ii), we will use Equation (ii') given by:

$$(ii') \quad \forall\, \mathbf{e} \in \mathbb{Z}_{q_\eta}^N, \prod_{i=1}^N (M \cdot \mathbf{e})_i = \prod_{i=1}^N e_i.$$

Therefore, the characterisation of permutation matrix is modelled in the CCSA logic by the following rule:

L.π:Charac

$$\frac{\mathcal{E}; \Theta; \Gamma \vdash M \cdot \mathbf{1} = \mathbf{1} \qquad \mathcal{E}; \Theta; \Gamma \vdash \mathbf{prod}_N\ (M \cdot X) - \mathbf{prod}_N\ X = \mathbf{0}}{\mathcal{E}; \Theta; \Gamma \vdash \mathbf{perm}_N\ M}$$

Equations (ii) and (ii') are equivalent thanks to the *Schwartz-Zippel* lemma ([20, 17]). This lemma states that, for $f_d \in \mathbb{Z}_{q_\eta}[X_1, \ldots, X_N]$ a non-zero multivariate polynomial of total degree $d \in \mathbb{N}$ over $\mathbb{Z}_{q_\eta}$ and for $\mathbf{e} \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N$ a vector chosen uniformly at random in the vector space $\mathbb{Z}_{q_\eta}^N$, then

$$\Pr_{\mathbf{e} \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N} \left[\ f_d(\mathbf{e}) = 0\ \right] \leqslant \frac{d}{q_\eta^N}.$$

This result can be model by the following rule:

L.Sz

$$\frac{\mathcal{E}; \Theta; \Gamma \vdash \phi_{\mathrm{fresh}}^{\mathbf{x}_0}(P) \qquad \mathcal{E}; \Theta; \Gamma \vdash P(\mathbf{x}_0) = 0}{\mathcal{E}; \Theta; \Gamma \vdash P = \mathbf{0}}$$

Finally, to show that matrix M was indeed used to shuffle the inputted list of ciphertexts, the second zero-knowledge proof shows the following equation for any $\mathbf{e} \in \mathbb{Z}_{q_\eta}^N$ chosen uniformly at random:

$$\exists\, u \in \mathbb{Z}_{q_\eta}, \mathbf{c}' \circledast (M \cdot \mathbf{e}) = \phi_{pk}(\mathbf{c} \circledast \mathbf{e}\,;\, u).$$

By studying the set $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$ given by

$$\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi} = \left\{\mathbf{e} \in \mathbb{Z}_{q_\eta}^N \mid \exists\, v \in \mathbb{Z}_{q_\eta}, \mathbf{c}' \circledast (M_\pi \cdot \mathbf{e}) = \phi_{pk}(\mathbf{c} \circledast \mathbf{e}\,;\, v)\right\},$$

we show the equivalence between the two following properties

1. There exists a vector of random values $\mathbf{r} = (r_i)_{i=1}^N \in \mathbb{Z}_{q_\eta}^N$ such that: $\forall\, i \in [\![1; N]\!], c'_{\pi(i)} = \phi_{pk}(c_i\,;\, r_i)$.

2. When vectors $\mathbf{e}$ of the vector space $\mathbb{Z}_{q_\eta}^N$ are chosen uniformly at random, we have: $\Pr_{\mathbf{e}\in\mathbb{Z}_{q_\eta}^N}\left[\ \mathbf{e} \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}\ \right] > \frac{1}{q_\eta}$.

This equivalence gives us the following rule to characterise the *shuffle* property of a mix-server:

L.Sfm:Charac

$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ \pi \qquad \mathcal{E};\Theta;\Gamma \vdash \phi^{\mathbf{e}}_{\text{fresh}}(\mathbf{c},\mathbf{c}',\pi) \qquad \mathcal{E};\Theta;\Gamma \vdash \exists\, v.\ \mathbf{c}' \circledast (\pi\cdot\mathbf{e}) = \mathbf{shuf\text{-}map}\ pk\ (\mathbf{c}\circledast\mathbf{e})\ v}{\begin{array}{c}\mathcal{E},(\mathbf{x}:\mathbf{msg});\Theta;\Gamma \vdash \exists\,(v_{\mathbf{x}}:\mathbf{rand}).\ \mathbf{c}'\circledast(\pi\cdot\mathbf{x}) = \\ \mathbf{shuf\text{-}map}\ pk\ (\mathbf{c}\circledast\mathbf{x})\ v_{\mathbf{x}}\end{array}}$$

4.3 Security properties

Formally, security properties for a mix-server are defined as follows. In both following properties, the commitment key parameter *ck* is honestly computed by the setup algorithm **gencomkey** and is publicly sent on the network to all agents.

- **(*Permutation secrecy*)** For this property, the mix-server behaves honestly while the verifier is controlled by an adversary $\mathcal{A}$. The idea of the secrecy property is to show that there is no way for the adversary $\mathcal{A}$ to guess the permutation used by the mix-server if the mix server behaves according to the protocol. To prove it, we ask the adversary to generate two permutations π_0 and π_1 in $\mathfrak{S}_N$ and send them to the mix-server. Then, the mix-server secretely chooses one of both permutations, depending on a secret random bit $\beta \in \{0,1\}$, and mixes the ballots with the permutation π_β. At this step, the adversary takes all the mix-server outputs and tries to guess the secret bit β. The adversary wins the *permutation secrecy game* $\text{Secrecy}^{\mathcal{A}}(1^\eta\,;\,\beta)$ if he correctly guessed the secret bit β. If he cannot win the game with significant probability, then we consider that the permutation secrecy is guaranteed. This property is modelled in the CCSA logic by the following property:

$$\mathcal{E};\varnothing \vdash \mathbf{mix}_{\phi_{pk}}\ \pi\ (ck\ n)\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c} \sim \mathbf{mix}_{\phi_{pk}}\ \mathrm{id}\ (ck\ n)\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}$$

- **(*Verifiability*)** For this property, the mix-server is controlled by an adversary $\mathcal{A}$ and the verifier is behaves honestly. Intuitively the verifiability property ensures that as long as the mix-server provides proofs that are accepted by the verifier, the decryption of the outputted list of ballots is a permutation of the decryption of the inputted one. That is, the adversary first outputs a vector $\mathbf{a} \in \mathbb{Z}_{q_\eta}^N$ along with a proof transcript $\mathbf{p}_\pi$ showing the relation $\mathcal{R}^{\text{com}}(\mathbf{e}_\pi)$ for some vector $\mathbf{e}_\pi \in \mathbb{Z}_{q_\eta}^N$ computed by the verifier $\mathcal{V}$. Then, the adversary outputs two ciphertexts lists $\mathbf{c},\mathbf{c}' \in \mathcal{C}_{\mathbb{CS}}^N$ of length N and a secret key $sk \in \mathbb{Z}_{q_\eta}$ along with a proof transcript $\mathbf{p}_\phi$ showing the relation $\mathcal{R}^{\text{shuffle}}_{\phi_{pk}}(\mathbf{e}_\phi)$ for some other vector $\mathbf{e}_\phi \in \mathbb{Z}_{q_\eta}^N$. The adversary wins the *verifiability game* $\text{Verif}^{\mathcal{A}}(1^\eta)$ when the proofs are accepted by the verifier, but the decryption of the outputted ciphertexts list $\mathbf{c}'$ leads to a different decryption of the inputted ciphertexts list $\mathbf{c}$. We state this property in the CCSA model by the following property

$$\mathcal{E};\varnothing \vdash \left[\begin{array}{ll} & \mathbf{zkp\text{-}verif}_\pi\ (ck\ n,\mathbf{a},\mathbf{e}_\pi)\ \langle\alpha_\pi,r_\pi,z_\pi\rangle \\ \wedge & \mathbf{zkp\text{-}verif}_\phi\ (ck\ n,\mathbf{a},\mathtt{pk}_{\mathbb{CS}}\ sk,\mathbf{c},\mathbf{c}',\mathbf{e}_\phi)\ \langle\alpha_\phi,r_\phi,z_\phi\rangle \\ \wedge & \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c} \\ & \rightarrow \\ & \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}' \\ \wedge & \mathbf{eqm}_N\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c})\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c}') \end{array}\right]$$

where $\mathbf{eqm}_N$ is the predicate standing for equality of lists in the multisets sense.

5 Proof of verifiability

In the case of the verifiability property, the adversary $\mathcal{A}$ *plays* the role of a mix-server and the verifier $\mathcal{V}$ is honest. This property is a trace property, *i.e.* at the very end of the mix-server protocol, we check whether or not the verifiability property holds for the obtained trace by considering all the messages exchanged between the adversary $\mathcal{A}$ and the verifier $\mathcal{V}$. More precisely, the full trace ϕ is given by $\phi \stackrel{\text{def}}{=} ck\ n, \mathbf{a}, \mathbf{e}_\pi, \alpha_\pi, r_\pi, z_\pi(r_\pi), sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi, \alpha_\phi, r_\phi, z_\phi(r_\phi)$ where $\mathbf{a}$, α_π, $z_\pi(r_\pi)$, sk, $\mathbf{c}$, $\mathbf{c}'$, α_ϕ and $z_\phi(r_\phi)$ are outputted by the adversary $\mathcal{A}$.

5.1 Sketch of verifiability proof

To prove the verifiability property, we first need to extract N witnesses for the correct commitment relation $\mathcal{R}^{\text{com}}(\mathbf{e}_i)$ for a vector basis $(\mathbf{e}_i)_{i=1}^N$ sends by the verifier to be able to rebuild the matrix π contained in the commit message $\mathbf{a} \in \mathbb{G}_{q_\eta}^N$ sends by the adversary. Then, by extracting one last witness for the correct commitment relation $\mathcal{R}^{\text{com}}(\mathbf{e})$ where $\mathbf{e}$ is a vector chosen uniformly at random and independently from the other vectors $(\mathbf{e}_i)_{i=1}^N$, we use the *binding* property of the commitment scheme $\mathbb{KS}(\mathbb{Z}_{q_\eta}^N)$ to show that π satisfies both of the following equations (i) $\pi \cdot \mathbf{1} = \mathbf{1}$ and (ii) $\mathbf{prod}_N\ (\pi \cdot X) = \mathbf{prod}_N\ X$ and hence conclude that π represents a permutation. Finally, from the second zero-knowledge proofs, extract a witness for the correct shuffle relation $\mathcal{R}_{\phi_{pk}}^{\text{shuffle}}(\mathbf{e})$ for a vector $\mathbf{e}$ chosen uniformly at random allows to show that both ciphertexts lists $\mathbf{c}$ and $\mathbf{c}'$ are link by the *shuffle-friendly* map ϕ_{pk}, *i.e.* for all $i \in [\![1; N]\!]$, we have the following property: $\exists\, r_i \in \mathbb{Z}_{q_\eta}, c'_{\pi(i)} = \phi_{\mathtt{pk}_{\mathbb{CS}}(sk)}(c_i\,;\, r_i)$. Those last equations implies, by the correctness of *shuffle-friendly* maps (*i.e.* apply the function ϕ_{pk} does not change the decryption of ciphertexts), the equality in the sense of multisets of the lists $\mathbf{dec\text{-}list}_{\mathbb{CS}}^{(N)}(sk, \mathbf{c})$ and $\mathbf{dec\text{-}list}_{\mathbb{CS}}^{(N)}(sk, \mathbf{c}')$ which is the property we want.

To be able to extract witnesses, the rewinding technique is mandatory. Roughly, this technique states that one can run the adversary $\mathcal{A}$ *twice*: the adversary $\mathcal{A}$ is run a first time, next we rewind her to a previous state, and finally the adversary is run a second time from this state.

$$\mathtt{st}_{\mathcal{A}}(t_0) \rightsquigarrow \mathtt{st}_{\mathcal{A}}(t) \rightsquigarrow \underbrace{\mathtt{st}_{\mathcal{A}}(t_f) \hookrightarrow \mathtt{st}_{\mathcal{A}}(t)}_{\text{rewinding}} \rightsquigarrow \mathtt{st}_{\mathcal{A}}(t'_f)$$

The rewinding argument is used in two different contexts. The first one is for the witness extraction from a Σ-protocol with the special-soundness property. As a remind, the idea behind the special-soundness property is to obtain two different proof transcripts but for the same commit message, only the challenge and the response differ in these two proof transcripts. The second one is when we rebuild the matrix commited in the vector $\mathbf{a}$, we have to obtain enough equations. If the first use case of rewinding can be abstract as a black-box, the second use case can not. Indeed, to be able to apply the system solver of linear equations **solve**, the family of vectors $(\mathbf{e}_i)_{i=1}^N$ used to extract witnesses, and then to get the linear equations system, has to be a free family. However, even if the probability for a vector family to be free is overwhelming, the verifier must generates more than N vectors because the adversary may not give an accepted proof transcript for all vectors produce by the verifier. As a matter of fact, the adversary $\mathcal{A}$ sort of *chooses* which vectors she will responds by an accepted proof.

5.2 Rewinding in the CCSA logic

To present how to model rewinding in the CCSA model, we take the witness extraction for Σ-protocols using the *special-soundness* axiom as enlightening example. Then, let $\mathcal{R}$ be a computable binary relation and let $\Sigma_{\mathcal{R}} = (\mathcal{S}, \mathcal{P}, \mathcal{V})$ be a Σ-protocol for the relation $\mathcal{R}$. As $\Sigma_{\mathcal{R}}$ is a Σ-protocol, $\Sigma_{\mathcal{R}}$ is in particular *special-sound*. Therefore there exists an extractor $\mathcal{E}_{\mathcal{R}}$ for this Σ-protocol which outputs a witness $w \in \mathcal{W}_{\mathcal{R}}$ for a statement $x \in \mathcal{X}_{\mathcal{R}}$ (*i.e.* we have $(x, w) \in \mathcal{R}$) on inputs two different proof transcripts but for the same commit message $\mathbf{p}_{\mathcal{R}}^{(i)}(c_i) \stackrel{\text{def}}{=} \langle \alpha, c_i, z_i(c_i) \rangle$ for $i = 1, 2$. We consider an adversary $\mathcal{A}$ which acts as of the prover $\mathcal{P}$. The procedure of witness extraction procedure consists in running the adversary on a different challenges

for the same commit message. The procedure can be found in Appendix C, Algorithm 1. This procedure is polynomial if the probability for the adversary $\mathcal{A}$ to produces a proof transcript accepted by the verifier is non-negligible.

To model this witness extraction procedure, a local aproach can not be adopted, because if the verifier $\mathcal{V}$ accepts some particular proof transcript, *i.e.* if $\mathcal{E};\Theta;\Gamma \vdash \mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ (p^{(0)}_{\mathcal{R}}, c, p^{(1)}_{\mathcal{R}}(c))$, it says nothing about whether or not $\mathcal{V}$ accepts another proof transcript for some other challenge $c' \neq c$. Said otherwise, we have

$$\mathcal{E};\Theta;\Gamma, c \neq c', \mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ (p^{(0)}_{\mathcal{R}}, c, p^{(1)}_{\mathcal{R}}(c)) \not\vdash \mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ (p^{(0)}_{\mathcal{R}}, c', p^{(1)}_{\mathcal{R}}(c'))$$

Therefore, we necessarily have to deal with global formulas. As seen before, we need to talk about global formulas which holds with non-negligible probability. In the CCSA logic, by definition of $[\phi]$ semantics, the property $\mathcal{E};\Theta \vdash \tilde{\neg}\,[\neg\phi]$ means that the following function is non-negligible

$$\eta \longmapsto \Pr_{\rho\in\Omega}\left[\ [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \right]$$

which is exactly the kind of properties we want to catch. However, we need explicit lower bound for this previous function, we therefore add a new predicate which gives us a lower bound for an event to occurs. We first define non-negligible parameter in the CCSA logic as terms $g : \mathbf{real}$ such that the property $\mathbf{non\text{-}negl}(g)$ holds. The non-negligible predicate **non-negl** holds for a real parameter $g : \mathbf{real}$ when the function $\eta \longmapsto \mathbb{E}_{\rho\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big)$ defines a non-negligible function. Therefore, for all formula $\phi : \tau_1 \to \ldots \to \tau_n \to \mathbf{bool}$ and for all non-negligible parameter $g : \mathbf{real}$ with $\mathbf{non\text{-}negl}(g)$, we define the global predicate ${}_g[\phi]$ with the following semantics

$$[\![{}_g[\phi]]\!]_{\mathbb{M}:\mathcal{E}} \overset{\text{def}}{=} \forall\,\eta \in \mathbb{N}^*, \Pr_\rho\left[\ [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \right] \geqslant \mathbb{E}_\rho\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

Notice that, as g is a non-negligible parameter, we indeed have:

G.$\tilde{\neg}$:Charac

$$\mathcal{E};\Theta \vdash \tilde{\neg}\,[\neg\,\phi] \mathrel{\tilde{\leftrightarrow}} \tilde{\exists}\ (g : \mathbf{real}).\ \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ {}_g[\phi]$$

Informally, we need a second property that states that from a property $\phi \overset{\text{def}}{=} \lambda(x : \tau).\ \phi\ x$ which holds with non-negligible probability, by resampling x enough times, we can make ϕ almost always true. Said differently, to return to our running example of witness extraction for Σ-protocols, we want that if the adversary $\mathcal{A}$ produces proof transcripts accepted by the verifier with non-negligible probability, then the function $\mathbf{extract\text{-}sigp}_{\mathcal{R}}$ produces a witness w for a statement x with overwhelming probability and in polynomial time. To the best of our knowledge, the previous work based on the CCSA logic defines the non-negligibility definition to capture *at least* the usual cryptographic definition of non-negligibility. In fact, the two properties we want on our new predicate ${}_g[\phi]$ means that the non-negligible definition in the CCSA logic must be *at most* the usual cryptographic definition of non-negligibility. Therefore, a term $g : \mathbf{real}$ is a non-negligible parameter (*i.e.* the property $\mathbf{non\text{-}negl}(g)$ holds) when there exists a polynomial P such that

$$\forall\,\eta \in \mathbb{N}^*, \exists\,\eta_0 > \eta, \mathbb{E}_\rho\big([\![g]\!]^{\eta_0,\rho}_{\mathbb{M}:\mathcal{E}}\big) > \frac{1}{P(\eta_0)}.$$

However, the new predicate ${}_g[\phi]$ of non-negligible formulas is not enough to successfully capture the rewinding technique. Indeed, if a formula $\mathcal{E};\Theta \vdash [\psi]$ holds, it means that the function

$$\eta \longmapsto \Pr_\rho\left[\ [\![\psi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \right]$$

is overwhelming. More precisely, this means that if ψ depends on random samplings given by the random tape ρ, *all* these samples are re-computed. While what the procedure given in Algorithm 1 means is that we have to rewind only a part of samples used to compute proof transcripts.

Therefore, if a formula ψ depends on random samplings $\omega \overset{\text{def}}{=} \{r_i\}_{i=1}^n$, we want to split these samplings in two parts $\omega_{\text{rewind}} \overset{\text{def}}{=} \{r_i\}_{i\in I}$ for *fixed* random samplings and $\omega_{\text{fix}} \overset{\text{def}}{=} \omega \setminus \omega_{\text{rewind}}$ for random samplings we want

to rewind where $I \subseteq [\![1; n]\!]$. To do so, we consider the property $\phi : \tau_1 \to \cdots \to \tau_p \to \mathbf{bool}$ where the parameter of ϕ are exactly the parameter of ψ we want to rewind, *i.e.* we have $p = \mathrm{Card}(I)$ and, for all security parameter $\eta \in \mathbb{N}^*$ and for all $i \in [\![1; p]\!]$, we have $r_i \in [\![\tau_i]\!]^\eta_{\mathbb{M}} \iff r_i \in \omega_{\text{rewind}}$. Therefore, given a property $\phi : \tau_1 \to \cdots \to \tau_p \to \mathbf{bool}$ and a parameter $g : \mathbf{real}$ with $\mathbf{non\text{-}negl}(g)$, we define the predicate **low-bound** with the following semantics

$$\forall\, \eta \in \mathbb{N}^*, \forall\, \rho \in \Omega, [\![\mathbf{low\text{-}bound}\ g\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \stackrel{\text{def}}{=} \Pr_{r_i \in [\![\tau_i]\!]^\eta_{\mathbb{M}}, i \in [\![1;p]\!]} \left[\ [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1, \ldots, r_p)\ \right] \geqslant \mathbb{E}_{\rho'}\big([\![g]\!]^{\eta,\rho'}_{\mathbb{M}:\mathcal{E}}\big).$$

Thus, the predicate **low-bound** capture exactly the probability that a formula ψ is non-negligible when part of random samplings, those which belongs to the set ω_{fix}, are fixed. Before properly states the rewinding lemma, we will states a crucial rule on the **low-bound** predicate, the one which eliminate this predicate.

$$\text{G.LB:Elim}\quad \frac{\mathcal{E}; \Theta \vdash \tilde{\forall}\, g : \mathbf{real}.\ \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g) \tilde{\to} [\mathbf{low\text{-}bound}\ g\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r}]}{\mathcal{E}; \Theta \vdash [\phi\ \mathbf{r} \to \psi\ \mathbf{r}]}$$

Proposition 5.1. The rule G.LB:Elim is sound.

Proof. Soundness of this rule is proved by contraposition. That is, we prove the property $\mathcal{E}; \Theta \vdash \tilde{\exists}\ (g : \mathbf{real}).\ \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g)$
$\tilde{\wedge}\ \tilde{\neg}\ [\mathbf{low\text{-}bound}\ g\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r}]$ under the hypothesis
$\mathcal{E}; \Theta \vdash \tilde{\neg}\ [\phi\ \mathbf{r} \to \psi\ \mathbf{r}]$. After using characterisation of non-negligible formulas G.$\tilde{\neg}$:Charac, it remains to used a key argument of the **low-bound** introduction rule given by

$$\text{G.LB:Intro}\quad \frac{\mathcal{E}; \Theta \vdash {}_{g}[\phi\ \mathbf{r}]}{\mathcal{E}; \Theta \vdash {}_{g/2}[\mathbf{low\text{-}bound}\ (g/2)\ \phi]}$$

More details can be found in Appendix B. □

Finally, we can now properly states the rewinding lemma.

Proposition 5.2 (Rewinding lemma)**.** Let $\phi \stackrel{\text{def}}{=} \lambda x.\phi(x) : \tau \to \mathbf{bool}$ [ptime] be a polynomial time property. Let $n \in \mathbb{N}^*$ be a natural number. Let $g : \mathbf{real}$ with $\mathbf{non\text{-}negl}(g)$ be a non-negligible parameter. Then we conclude the following judgement.

$$\mathcal{E}; \Theta \vdash \tilde{\exists}\, k : \mathbf{nat}.\ \mathbf{det}(k)\ \tilde{\wedge}\ \mathbf{pbound}(k) \tilde{\to} \tilde{\exists}\, \mathbf{r}_s : \mathbf{nat} \to \tau. \tilde{\exists}\, \mathbf{select}^{(n)}_{\text{rand}} : \mathbf{nat} \to (\mathbf{nat} \to \tau) \to \mathbf{set}_n(\tau).$$
$$[\mathbf{low\text{-}bound}\ g\ \phi \to \forall\, (\mathbf{r}_s\ t : \tau) \in \mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s.\ \phi\ (\mathbf{r}_s\ t)] \tilde{\wedge}\ [\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]$$

To prove the rewinding lemma, we define an adversarial selection function $\mathbf{select}^{(n)}_{\text{rand}} : \mathbf{nat} \to (\mathbf{nat} \to \tau) \to \mathbf{set}_n(\tau)$, its complexity study gives a concrete value for the natural term $k : \mathbf{nat}$ and which satisfies verifies both predicates $\mathbf{det}(k)$ and $\mathbf{pbound}(k)$. The full definition can be found in Appendix C.

To end our enlightening example of witness extraction for Σ-protocols, the application of the rewinding lemma gives us existency of a polynomial bounded and deterministic natural number $k : \mathbf{nat}$, a source of random challenges $\mathbf{r}_s : \mathbf{nat} \to \mathbf{msg}$ and an adversarial selection function $\mathbf{select}^{(2)}_{\text{chall}} : \mathbf{nat} \to (\mathbf{nat} \to \mathbf{msg}) \to \mathbf{set}_2(\mathbf{msg})$ such that

$$\mathcal{E}; \Theta \vdash [\mathbf{low\text{-}bound}\ g\ \psi_{\mathcal{R}} \to \forall\, (\mathbf{r}_s\ t : \mathbf{msg}) \in \mathbf{select}^{(2)}_{\text{chall}}\ k\ \mathbf{r}_s.\ \psi_{\mathcal{R}}\ (\mathbf{r}_s\ t)] \tilde{\wedge}\ [\mathbf{select}^{(2)}_{\text{chall}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]$$

where $g : \mathbf{real}$ is a non-negligible and deterministic parameter and $\psi_{\mathcal{R}}$ is the property defined by

$$\psi_{\mathcal{R}} \stackrel{\text{def}}{=} \lambda r.\ \mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ \langle \alpha, r, z(r) \rangle.$$

Therefore, as the property $\mathbf{select}^{(2)}_{\text{chall}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}$ holds, this leads to the existence of two challenge terms $r_1, r_2 : \mathbf{msg}$ with $r_1 \neq r_2$ and $\psi_{\mathcal{R}}\ r_i$. Hence, we have all gathered hypothesis to use the *special-soundness* axiom L.Σ-P:SpSound and conclude

$$\mathcal{E}; \Theta \vdash [\mathbf{low\text{-}bound}\ g\ \psi_{\mathcal{R}} \to \psi_{\mathcal{R}}\ r \to \mathbf{zkp\text{-}rel}_{\mathcal{R}}\ x\ (\mathbf{zkp\text{-}extract}_{\mathcal{R}}\ x\ \mathbf{p}^{(1)}_{\mathcal{R}}(r_1)\ \mathbf{p}^{(2)}_{\mathcal{R}}(r_2))]$$

where $\mathbf{select}^{(2)}_{\text{chall}}\ k\ \mathbf{r}_s = \{r_1, r_2\}$ and $\mathbf{p}^{(i)}_{\mathcal{R}}(r_i) = \langle \alpha, r_i, z(r_i)\rangle$. As this property holds for all non-negligible and deterministic parameter $g : \mathbf{real}$, we conclude by the elimination rule G.LB:Elim, the following property

$$\mathcal{E}; \Theta \vdash [\mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ \langle \alpha, r, z(r)\rangle \to \mathbf{zkp\text{-}rel}_{\mathcal{R}}\ x\ (\mathbf{zkp\text{-}extract}_{\mathcal{R}}\ x\ \mathbf{p}^{(1)}_{\mathcal{R}}(r_1)\ \mathbf{p}^{(2)}_{\mathcal{R}}(r_2))].$$

Therefore, this previous property fully model the witness extraction for Σ-protocols and use an adversarial selection function $\mathbf{select}^{(2)}_{\text{chall}}$ defined in Algorithm 2.

Property transfer by adversarial selection

G.Sel

$$\frac{\begin{array}{c}\mathcal{E} \vdash \mathbf{r}_s : \mathbf{nat} \to \tau \qquad \mathcal{E} \vdash \mathbf{select}^{(n)}_{\text{rand}} : \mathbf{nat} \to (\mathbf{nat} \to \tau) \to \mathbf{set}_n(\tau) \\ \mathcal{E}; \Theta \vdash \mathbf{det}(k)\ \tilde{\wedge}\ \mathbf{pbound}(k) \qquad \mathcal{E}; \Theta \vdash [\phi\ (\mathbf{r}_s\ 1)\ \ldots\ (\mathbf{r}_s\ n)] \qquad \mathcal{E}; \Theta \vdash [\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s \subset \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]\end{array}}{\mathcal{E}; \Theta \vdash [n \leqslant k \to \phi\ (\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s)]}$$

Intuitively, this property states that is a property ϕ holds with overlwhelming probability over a set a random samplings, then it still holds even if the adversary is allowed to select the randomness from a polynomially sized set. Intuitively, this property follows from the fact that in a polynomially sized set of randomness, the probability of finding a subset that invalidates ϕ is negligible. As a short example $[\mathbf{basis}_N\ (\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s)]$ holds, meaning that even if the adversary can select randomness in a polynomially size set, N random vectors still form a basis with overwhelming probability.

5.3 Verifiability proof

Let $\phi \stackrel{\text{def}}{=} ck\ n, \mathbf{a}, \mathbf{e}_\pi, \alpha_\pi, r_\pi, z_\pi(r_\pi), sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi, \alpha_\phi, r_\phi, z_\phi(r_\phi)$ be a trace such that

$$\mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_\pi)\ \langle \alpha_\pi, r_\pi, z_\pi(r_\pi)\rangle \wedge\ \mathbf{zkp\text{-}verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r_\phi, z_\phi(r_\phi)\rangle \wedge\ \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}.$$

5.3.1 Extraction of the commited matrix

To be able to rebuild the commited matrix, we have to extract N witnesses $(\mathbf{e}'_i, k_i)_{i=1}^N$ for the relations of correct commitment $\mathcal{R}^{\text{com}}(\mathbf{e}_i)$, where $(\mathbf{e}_i)_{i=1}^N$ is a free family of $\mathbb{Z}_{q_\eta}^N$. Consequently, there is two steps of rewinding, one on the vectors $\mathbf{e}_i$, $i \in [\![1; N]\!]$, and the other one is once we obtain a candidate vector $\mathbf{e}_i$, we have to rewind the challenge $r \in \mathbb{Z}_{q_\eta}$ to be able to use the *special-soundness* axiom. Therefore, in that case, we have to use two times the predicate **low-bound**, one states there is enough random vectors to rewind and the second one states that for a chosen vector, there is enough random challengs to rewind. Hence, if we denote by ψ_π the formula

$$\psi_\pi \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \lambda r.\ \mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r)\rangle,$$

we have to suppose the following property

$$\mathbf{low\text{-}bound}\ g\ (\lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e}))$$

for any two non-negligible and deterministic parameters $g, g' : \mathbf{real}$.

Consequently, by two successives rewinding lemma application, we obtain existence of two random source names $\mathbf{e}_s : \mathbf{nat} \to \mathbf{msg}$ for vectors and $\mathbf{r}_s : \mathbf{nat} \to \mathbf{msg}$ for challenges such that (i) $\mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s =$

$\{\mathbf{e}_s\ t_i\}_{i=1}^N$, (ii) $\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s = \{\mathbf{r}_s\ r_{i,1}, \mathbf{r}_s\ r_{i,2}\}$ with $t_1, \ldots, t_N : \mathbf{nat}$ pairwise distincts and $r_{i,1} \neq r_{i,2}$ and (iii) $\psi_\pi\ (\mathbf{e}_s\ t_i)\ (\mathbf{r}_s\ r_{i,j})$ holds for all $i \in [\![1; N]\!]$ and $j \in \{1, 2\}$. Therefore, for all $i \in [\![1; N]\!]$, we have $\psi_\pi\ (\mathbf{e}_s\ t_i)\ (\mathbf{r}_s\ r_{i,1})$ and $\psi_\pi\ (\mathbf{e}_s\ t_i)\ (\mathbf{r}_s\ r_{i,2})$ with $r_{i,1} \neq r_{i,2}$. Thus, by the *special-soundness* axiom, we obtain N witnesses $w_\pi(i)$, for all $i \in [\![1; N]\!]$, defined by

$$w_\pi(i) \stackrel{\text{def}}{=} \mathbf{zkp\text{-}extract}_\pi\ (x_\pi(i))\ (\mathbf{p}_\pi^{(i,1)}(\mathbf{r}_s\ r_{i,1}))\ (\mathbf{p}_\pi^{(i,2)}(\mathbf{r}_s\ r_{i,2}))$$

where $x_\pi(i) \stackrel{\text{def}}{=} (ck\ n, \mathbf{a}, \mathbf{r}_s\ t_i)$ and $\mathbf{p}_\pi^{(i,j)}(c_j) \stackrel{\text{def}}{=} \langle \alpha_\pi(i), c_j, z_\pi(i, c_j)\rangle$ such that $w_\pi(i)$ verifies the property $\mathbf{zkp\text{-}rel}_\pi\ (x_\pi(i))\ (w_\pi(i))$. In particular, by using the function **solve**, we conclude the existency of two terms M and $\mathbf{s}$ such that $\mathbf{a}$ is a commit message to the matrix M, *i.e.* such that $\mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s}$.

5.3.2 M represents a permutation

Next, we want to apply the characterisation of permutation matrix to show that the matrix M represents a permutation. To do so, we have to extract a new witness to the relation of correct commitment $\mathcal{R}^{\text{com}}(\mathbf{e}_\pi)$ where the vector $\mathbf{e}_\pi$ must be independent from the matrix M in order to apply the Schwartz-Zippel lemma and conclude. Therefore, once again we suppose that we have enough of random challenges to rewind with the vector $\mathbf{e}_\pi$, *i.e.* we suppose the following property

$$\mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e}_\pi).$$

Therefore, with this new witness, we will be able to verify conditions of the characterisation of permutation matrix and show that M is indeed a permutation matrix.

5.3.3 M was used to shuffle the inputted ciphertexts list with the *shuffle-friendly* map ϕ_{pk}

Finally, after have rebuilt a matrix M such that $\mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s}$ for some vector $\mathbf{s}$ and have shown that M is a permutation matrix, we can verify if the outputted ciphertexts list $\mathbf{c}'$ is the *shuffle* of the inputted ciphertexts list $\mathbf{c}$ using the permutation matrix M and the *shuffle-friendly* map ϕ_{pk}. Once again, to be able to show this property, we have to extract a witness but from the second zero-knowledge proof this time. Therefore, we define by ψ_ϕ the following formula

$$\psi_\phi \stackrel{\text{def}}{=} \lambda r.\ \mathbf{zkp\text{-}verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r, z_\phi(r)\rangle$$

and we suppose there is enough of random challenges to be able to rewind and then extract a witness. Thus, we suppose the following property $\mathbf{low\text{-}bound}\ g'\ \psi_\phi$. As soon as the witness is obtained, we can prove that M was used to shuffle the inputted ciphertexts list.

5.3.4 Proof of the verifiability property

We denote by $\mathcal{H}$ the function defined by

$$\mathcal{H} \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \lambda r.\ \lambda r'.\mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r)\rangle \wedge \mathbf{zkp\text{-}verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r', z_\phi(r')\rangle \wedge \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}.$$

We want to prove the following formula

$$[\mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \mathbf{eqm}_N\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c})\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c}')]$$

As $\mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \psi_\pi\ \mathbf{e}_\pi\ r_\pi$ and $\mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \psi_\phi\ r_\phi$, we can use the 3 previous results to conclude the following property for all deterministic non-negligible parameter $g, g' : \mathbf{real}$:

$$\mathbf{low\text{-}bound}\ g\ (\lambda \mathrm{e}.\ \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e})) \to \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e}_\pi) \to \mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \mathbf{eqm}_N\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c})\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c}')$$

Therefore, by two applications of the elimination rule of predicate **low-bound** (one with the parameter g then another one with the parameter g'), we obtain the property we want to show.

6 Proof of permutation secrecy

To ease notations, we denote by $x_\pi(\pi)$ the statement $x_\pi(\pi) \stackrel{\text{def}}{=} (ck\ \ n, \mathbf{a}_\pi, \mathbf{e}_\pi)$ and $x_\phi(\pi)$ the statement $x_\phi(\pi) \stackrel{\text{def}}{=} (ck\ \ n, \mathbf{a}_\pi, (\mathtt{pk}_{\mathbb{CS}}\ \ sk), \mathbf{c}, \mathbf{c}'_\pi, \mathbf{e}_\phi)$. By unfolding the definition of the mix predicate $\mathbf{mix}_{\phi_{pk}}$, one has to prove the following indistinguishability

$$\mathcal{E}; \varnothing \vdash \mathbf{a}_\pi, \mathbf{p}_\pi(\pi), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi, \mathbf{p}_\phi(\pi) \sim \mathbf{a}_{\mathrm{id}}, \mathbf{p}_\pi(\mathrm{id}), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}, \mathbf{p}_\phi(\mathrm{id})$$

where $\mathbf{a}_\pi \stackrel{\text{def}}{=} \mathbf{com\text{-}mat}\ (ck\ n)\ \pi\ (\mathbf{s}\ i_1)$, $\mathbf{p}_\pi(\pi) \stackrel{\text{def}}{=} \mathbf{zkp\text{-}prove}_\pi\ x_\pi(\pi)\ w_\pi\ (r_\pi\ i_3)$, $\mathbf{c}'_\pi \stackrel{\text{def}}{=} \mathbf{shuffle}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ \pi\ (\mathbf{r}\ i_4)$ and $\mathbf{p}_\phi(\pi) \stackrel{\text{def}}{=} \mathbf{zkp\text{-}prove}_\phi\ x_\phi(\pi)\ w_\phi\ (r_\phi\ i_6)$. To prove this secrecy property, the first step is to use the *honest-verifier zero-knowledge* property for both zero-knowledge proofs. Indeed, both proof transcripts depend on statements where messages $\mathbf{a}_\pi$ and $\mathbf{c}'_\pi$ might appear. Therefore, we could not prove indistinguishability between both traces without use the *honest-verifier zero-knowledge* property to transform honest executions with $\mathbf{zkp\text{-}prove}_\mathcal{R}$ (which depends on witnesses) to simulated executions with $\mathbf{zkp\text{-}sim}_\mathcal{R}$ (which does not depend on witnesses). Hence, by applying this property to both proof transcripts, we have to prove the following property:

$$\begin{aligned}\mathcal{E}; \varnothing \vdash \mathbf{a}_\pi, \mathbf{zkp\text{-}sim}_\pi\ x_\pi(\pi)\ (r_\pi\ i_3), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v \\ \mathbf{then}\ \mathbf{c}'_\pi, \mathbf{zkp\text{-}sim}_\phi\ x_\phi(\pi)\ (r_\phi\ i_6) \sim \mathbf{a}_{\mathrm{id}}, \mathbf{zkp\text{-}sim}_\pi\ x_\pi(\pi)\ (r_\pi\ i_3), \\ \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}, \mathbf{zkp\text{-}sim}_\phi\ x_\phi(\pi)\ (r_\phi\ i_6)\end{aligned}$$

Next, using the function application rule G.∼:FA, by the elimination of duplicates with G.∼:Dup, by the elimination of fresh randoms with G.∼:Fresh and because terms computed by the adversary can be remove (thanks to G.∼:FA), it remains to prove the following indistinguishability:

$$\mathcal{E}; \varnothing \vdash \mathbf{a}_\pi, \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi \sim \mathbf{a}_{\mathrm{id}}, \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}$$

Finally, by using hiding property (rule G.Com:Hide) for the commitment scheme for matrix $\mathbb{KS}(\mathbb{Z}_{q_\eta}^{N\times N})$ and the indistinguishability of $\phi_{\mathtt{pk}_{\mathbb{CS}}(sk)}$ outputs property (rule G.Sfm:IndCPA), we achieve the secrecy proof.

7 Conclusion

Many electronic voting protocols use mixnets, which are critical to achieve security properties but unfortunately really hard to handle in automatic formal proofs frameworks. In this paper we propose a complete proof of Terelius-Wikström mixnet protocol in the CCSA logic. To do so, we introduce new predicates, rules and axioms in the logic to be able to handle zero-knowledge proofs and rewinding. To our knowledge, it is the first time that this protocol can be proved in a logical framework, and the first fully precise cryptographic proof of the Terelius-Wikström mixnet. As future work, we plan to include this new material in Squirrel, the tool implementing the CCSA logic. This will open the way to complete proofs of electronic voting protocols using mixnets. In parallel, this will also open the way to proofs of other kinds of protocols needing to handle zero-knowledge or rewinding. Another future improvement will be to adapt our work to handle non-interactive zero-knowledge proofs. This will imply to manage Fiat-Shamir transform, where reprogrammable Random Oracle Model is a prerequisite. This last primitive can be proven by using techniques developped to catch the rewinding lemma in the CCSA logic as it involves similar probabilistic arguments.

Aknowledgments

This work received funding from the France 2030 program managed by the French National Research Agency under grant agreement No. ANR-22-PECY-0006.

References

[1] Arapinis, M., Cortier, V., and Kremer, S. When are three voters enough for privacy properties? In *ESORICS (2)* (2016), vol. 9879 of *Lecture Notes in Computer Science*, Springer, pp. 241–260.

[2] Baelde, D., Delaune, S., Jacomme, C., Koutsos, A., and Moreau, S. An interactive prover for protocol verification in the computational model. In *SP* (2021), IEEE, pp. 537–554.

[3] Baelde, D., Koutsos, A., and Lallemand, J. A higher-order indistinguishability logic for cryptographic reasoning. In *LICS* (2023), pp. 1–13.

[4] Bana, G., and Comon-Lundh, H. A computationally complete symbolic attacker for equivalence properties. In *CCS* (2014), ACM, pp. 609–620.

[5] Barthe, G., Grégoire, B., and Béguelin, S. Z. Formal certification of code-based cryptographic proofs. In *POPL* (2009), ACM, pp. 90–101.

[6] Barthe, G., Grégoire, B., Heraud, S., and Béguelin, S. Z. Computer-aided security proofs for the working cryptographer. In *CRYPTO* (2011), vol. 6841 of *Lecture Notes in Computer Science*, Springer, pp. 71–90.

[7] Blanchet, B. An efficient cryptographic protocol verifier based on prolog rules. In *CSFW* (2001), IEEE Computer Society, pp. 82–96.

[8] Blanchet, B. A computationally sound mechanized prover for security protocols. *IEEE Trans. Dependable Secur. Comput. 5*, 4 (2008), 193–207.

[9] Cortier, V., Dragan, C. C., Dupressoir, F., and Warinschi, B. Machine-checked proofs for electronic voting: Privacy and verifiability for belenios. In *CSF* (2018), IEEE Computer Society, pp. 298–312.

[10] Cortier, V., Gaudry, P., and Glondu, S. Belenios: A simple private and verifiable electronic voting system. In *Foundations of Security, Protocols, and Equational Reasoning* (2019), vol. 11565 of *Lecture Notes in Computer Science*, Springer, pp. 214–238.

[11] Firsov, D., and Unruh, D. Reflection, rewinding, and coin-toss in easycrypt. In *CPP* (2022), ACM, pp. 166–179.

[12] Haenni, R., Koenig, R. E., Locher, P., and Dubuis, E. Chvote system specification. *IACR Cryptol. ePrint Arch.* (2017), 325.

[13] Haines, T., Goré, R., and Sharma, B. Did you mix me? formally verifying verifiable mix nets in electronic voting. In *SP* (2021), IEEE, pp. 1748–1765.

[14] Haines, T., Goré, R., and Tiwari, M. Machine-checking multi-round proofs of shuffle: Terelius-wikstrom and bayer-groth. In *USENIX Security Symposium* (2023), USENIX Association, pp. 6471–6488.

[15] Meier, S., Schmidt, B., Cremers, C., and Basin, D. A. The TAMARIN prover for the symbolic analysis of security protocols. In *CAV* (2013), vol. 8044 of *Lecture Notes in Computer Science*, Springer, pp. 696–701.

[16] Scerri, G., and Stanley-Oakes, R. Analysis of key wrapping apis: Generic policies, computational security. In *CSF* (2016), IEEE Computer Society, pp. 281–295.

[17] Schwartz, J. T. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM 27*, 4 (1980), 701–717.

[18] Terelius, B., and Wikström, D. Proofs of restricted shuffles. In *AFRICACRYPT* (2010), vol. 6055 of *Lecture Notes in Computer Science*, Springer, pp. 100–113.

[19] Wikström, D. A commitment-consistent proof of a shuffle. In *ACISP* (2009), vol. 5594 of *Lecture Notes in Computer Science*, Springer, pp. 407–421.

[20] Zippel, R. Probabilistic algorithms for sparse polynomials. In *EUROSAM* (1979), vol. 72 of *Lecture Notes in Computer Science*, Springer, pp. 216–226.

A Cryptographic definitions

$\mathrm{Hiding}^{\mathcal{A}_{setup},\mathcal{A}_{guess}}_{\mathbb{KS}(\mathcal{M})}\big(1^\eta\,;\,\beta\big)$ – Hiding property

$ck \leftarrow \mathrm{Gen}_{\mathcal{M}}(1^\eta)$; $r \xleftarrow{\$} \mathcal{R}^{\mathrm{COMMIT}}_{\mathcal{M}}$;
$(m_0, m_1) \leftarrow \mathcal{A}_{setup}(ck)$;
$c_\beta \leftarrow \mathrm{Com}_{\mathcal{M}}(ck, m_\beta\,;\,r)$;
$b \leftarrow \mathcal{A}_{guess}(c_\beta)$;
return b.

Figure 1: Cryptographic game of hiding

$\mathrm{Binding}^{\mathcal{A}}_{\mathbb{KS}(\mathcal{M})}\big(1^\eta\big)$ – Binding property

$ck \leftarrow \mathrm{Gen}_{\mathcal{M}}(1^\eta)$;
$(m_1, r_1) \leftarrow \mathcal{A}(ck)$; $(m_2, r_2) \leftarrow \mathcal{A}(ck)$;
$a_1 \leftarrow \mathrm{Com}_{\mathcal{M}}(m_1, r_1)$; $a_2 \leftarrow \mathrm{Com}_{\mathcal{M}}(m_2, r_2)$;
if $(m_1 \neq m_2 \wedge a_1 = a_2)$ **then** $b \leftarrow 1$ **else** $b \leftarrow 0$;
return b.

Figure 2: Cryptographic game of binding

$\text{Ind-CPA}^{\mathcal{A}}_{\phi_{pk},\,\boldsymbol{valid}}\big(\eta\,;\,\beta\big)$ – Output indistinguishability

$(sk, pk) \leftarrow \mathrm{KeyGen}_{\mathbb{CS}}(1^\eta)$; $r \xleftarrow{\$} \mathbb{Z}_{q_\eta}$;
$\big(c_0, c_1, v\big) \leftarrow \mathcal{A}_{setup}(\sigma, pk)$;
if $\big(\neg\,\big(\mathbf{valid}\ (c_0, c_1)\ v\big)\big)$ **then abort** ;
$c'_\beta \leftarrow \phi_{pk}(c_\beta\,;\,r)$;
$b \leftarrow \mathcal{A}_{guess}(\mathtt{st}_{\mathcal{A}}, c'_\beta)$;
return b.

Figure 3: Indistinguishability of ϕ_{pk} output game

$\mathrm{Secrecy}^{\mathcal{A}_{setup},\mathcal{A}_{guess}}\left(1^\eta\,;\,\beta\right)$ – Permutation secrecy property

$ck \leftarrow \mathrm{Gen}_{\mathcal{M}}(1^\eta)$; $\sigma \leftarrow \mathcal{S}_{ZK}(1^\eta, N)$; $(sk, pk) \leftarrow \mathrm{KeyGen}_{\mathbb{CS}}(1^\eta)$;

$\left(\begin{array}{l} \mathtt{st}_{\mathcal{A}}, (\mathbf{c}, v), (\pi_0, \pi_1), \\ (\mathbf{e}_\pi, r_\pi), (\mathbf{e}_\phi, r_\phi) \end{array}\right) \leftarrow \mathcal{A}_{setup}(\sigma, ck, pk)$;

$(\{\pi_\beta, \mathbf{s}\}, \mathbf{a}_{\pi_\beta}) \leftarrow \mathcal{M}_{mix}^{(\mathrm{off})}(\sigma, ck\,;\,\pi_\beta)$;

$\mathbf{p}_\pi \leftarrow \mathcal{P}_{mix}^{(\mathrm{off})}(\{\pi_\beta, \mathbf{s}\}, \sigma, \mathbf{e}_\pi\,;\,r_\pi)$;

$(\{\pi_\beta, \mathbf{s}, \mathbf{r}\}, \mathbf{c'}_{\pi_\beta}) \leftarrow \mathcal{M}_{mix}^{(\mathrm{on})}(\{\pi_\beta, \mathbf{s}\}, pk, \mathbf{c})$;

$\mathbf{p}_\phi \leftarrow \mathcal{P}_{mix}^{(\mathrm{on})}(\{\pi_\beta, \mathbf{s}, \mathbf{r}\}, \sigma, \mathbf{e}_\phi\,;\,r_\phi)$;

$b \leftarrow \mathcal{A}_{guess}(\mathtt{st}_{\mathcal{A}}, \mathbf{a}_{\pi_\beta}, \mathbf{p}_\pi, \mathbf{p}_\phi)$;

return b.

Figure 4: Cryptographic game of permutation secrecy for a mix-server

$\mathrm{Verif}^{\mathcal{A}_{setup},\mathcal{A}_{proofs}}\left(1^\eta\right)$ – Verifiability property

$\sigma \leftarrow \mathcal{S}_{ZK}(1^\eta, N)$; $ck \leftarrow \mathrm{Gen}_{\mathbb{Z}_{q_\eta}^{N\times N}}(1^\eta, N)$;

$\mathbf{e}_\pi \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N$; $\mathbf{e}_\phi \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N$;

$(\mathtt{st}_{\mathcal{A}}, \mathbf{a}, (sk, \mathbf{c}, \mathbf{c'})) \leftarrow \mathcal{A}_{setup}(\sigma, ck, N)$;

$b_\pi \leftarrow (\mathcal{A}_{proofs} \rightleftharpoons^{(\Sigma)}_{\mathcal{R}^{\mathrm{com}}(\mathbf{e}_\pi)} \mathcal{V}_\pi)((ck, \mathbf{a}))$;

$b_\phi \leftarrow (\mathcal{A}_{proofs} \rightleftharpoons^{(\Sigma)}_{\mathcal{R}^{\mathrm{shuffle}}_{\phi_{pk}}(\mathbf{e}_\phi)} \mathcal{V}_\phi)((ck, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}(sk), \mathbf{c}, \mathbf{c'}))$;

if $(\neg\, b_\pi \vee \neg\, b_\phi)$ **then return** 0 ;

if $\left(\begin{array}{ll} \mathtt{equal\_multisets} & (\mathtt{decrypt\_list}\ sk\ \mathbf{c}) \\ & (\mathtt{decrypt\_list}\ sk\ \mathbf{c'}) \end{array}\right)$ **then return** 0 ;

else return 1 ;

Figure 5: Cryptographic game of verifiability for a mix-server

B Proof system

Global and local judgements about logical reasoning can be found in [3] (in particular, proofs of soundness for these rules), but in Fig. 6 we remind several useful rules used in this paper.

B.1 Soundness of low-bound rules

In this subsection, we prove the soundness of rules about **low-bound** predicate. Let g : **real** with **non-negl**(g) be a non-negligible parameter. Let $\phi : \tau_1 \to \cdots \to \tau_n \to$ **bool** be a formula with n parameters.

- (G.LB:Elim) For this proof, we suppose also g to be deterministic, *i.e.* property **det**(g) holds. We proceed by contraposition, *i.e.* we suppose $\mathcal{E}; \Theta \vdash \tilde{\neg}\, [\phi\ \mathbf{r} \to \psi\ \mathbf{r}]$. Hence, by classical logic operations, we have $\mathcal{E}; \Theta \vdash \tilde{\neg}\, [\neg\ (\phi\ \mathbf{r} \wedge \neg\ (\psi\ \mathbf{r}))]$. Therefore, by characterisation of non-negligibility, we conclude by the rule G.$\tilde{\neg}$:Charac the existency of a non-negligible parameter g : **real** with **non-negl**(g) such that $\mathcal{E}'; \Theta \vdash_{g}[(\phi \wedge \neg\, \psi)\ \mathbf{r}]$ where $\mathcal{E}' \overset{\mathrm{def}}{=} \mathcal{E} \cup \{(g : \mathbf{real})\}$. Actually, without loss of generality, we can suppose that parameter g is deterministic, *i.e.* the property **det**(g) holds. Next, by the introduction rule G.LB:Intro of the predicate **low-bound**, we conclude
  $\mathcal{E}'; \Theta \vdash_{g/2}[\mathbf{low\text{-}bound}\ (g/2)\ (\phi \wedge \neg\, \psi)]$. As parameter g is deterministic and by the rule G.LB:Out, we conclude
  $$\mathcal{E}'; \Theta \vdash_{g^2/4}[\mathbf{low\text{-}bound}\ (g/2)\ (\phi \wedge \neg\, \psi) \wedge (\phi \wedge \neg\, \psi)\ \mathbf{r}].$$
  Therefore, as we have $\mathcal{E}'; \Theta \vdash_{1}[(\phi \wedge \neg\, \psi)\ \mathbf{r} \to \phi\ \mathbf{r}]$, we conclude by the global transitivity rule G.LB:Trans the following property
  $$\mathcal{E}'; \Theta \vdash_{g^2/4}[\mathbf{low\text{-}bound}\ (g/2)\ \phi \wedge (\phi \wedge \neg\, \psi)\ \mathbf{r}].$$

Global judgements: equivalence rules

$$\text{G.}\sim\text{:Refl}\quad \mathcal{E};\Theta \vdash \mathbf{u} \sim \mathbf{u}$$

$$\text{G.}\sim\text{:CS}\quad \frac{\mathcal{E};\Theta \vdash \mathbf{u}_l, b_l, s_l \sim \mathbf{u}_r, b_r, s_r \qquad \mathcal{E};\Theta \vdash \mathbf{u}_l, b_l, t_l \sim \mathbf{u}_r, b_r, s_r}{\mathcal{E};\Theta \vdash \mathbf{u}_l, \textbf{if } b_l \textbf{ then } s_l \textbf{ else } t_l \sim \mathbf{u}_r, \textbf{if } b_r \textbf{ then } s_r \textbf{ else } t_r}$$

$$\text{G.}\sim\text{:FA}\quad \frac{\mathcal{E};\Theta \vdash \mathbf{u}, t_l \sim \mathbf{u}, t_r \qquad \mathcal{E};\Theta \vdash \mathbf{adv}(f)}{\mathcal{E};\Theta \vdash \mathbf{u}, f\ t_l \sim \mathbf{u}, f\ t_r}$$

$$\text{G.}\sim\text{:Dup}\quad \frac{\mathcal{E};\Theta \vdash \mathbf{u}, t_l \sim \mathbf{u}, t_r}{\mathcal{E};\Theta \vdash \mathbf{u}, t_l, t_l \sim \mathbf{u}, t_r, t_r}$$

$$\text{G.}\sim\text{:Fresh}\quad \frac{\mathcal{E};\Theta \vdash [\phi^{n,t}_{\text{fresh}}(\mathbf{u}, t)]}{\mathcal{E};\Theta \vdash \mathbf{u}, n\ t \sim \mathbf{u}, n_{\text{fresh}}\ ()}$$

Other rules

$$\text{L.ByGlob}\quad \frac{\mathcal{E};\Theta \vdash [\phi]}{\mathcal{E};\Theta;\Gamma \vdash \phi}$$

$$\text{L.Rewrite}\quad \frac{\mathcal{E};\Theta;\Gamma \vdash \phi[s] \qquad \mathcal{E};\Theta;\Gamma \vdash s = t}{\mathcal{E};\Theta;\Gamma \vdash \phi[t]}$$

$$\text{G.R-}\tilde{\exists}\quad \frac{\mathcal{E};\Theta \vdash F\,\{x \longmapsto t\} \qquad \mathcal{E} \vdash (t : \tau)}{\mathcal{E};\Theta \vdash \tilde{\exists}\,(x : \tau).\ F}$$

$$\text{L.Comp}\quad \frac{}{\mathcal{E};\Theta;\Gamma \vdash \textbf{if } b \textbf{ then } \langle t_1, \ldots, t_n\rangle = \langle \textbf{if } b \textbf{ then } t_1, \ldots, \textbf{if } b \textbf{ then } t_n\rangle}$$

$$\text{G.}\tilde{\neg}\text{:Charac}\quad \frac{}{\mathcal{E};\Theta \vdash \tilde{\neg}\,[\neg\,\phi] \Leftrightarrow \tilde{\exists}\,(g : \textbf{real}).\ \textbf{non-negl}(g)\ \tilde{\wedge}\ {}_g[\phi]}$$

Figure 6: Used structural local and global rules

However, by logical operations, we have

$$\begin{aligned}
&\textbf{low-bound}\ (g/2)\ \phi \wedge (\phi \wedge \neg\,\psi)\ \mathbf{r}\\
&\quad = \quad \textbf{low-bound}\ (g/2)\ \phi \wedge \neg\,(\neg\,\phi \vee \psi)\ \mathbf{r}\\
&\quad = \quad \textbf{low-bound}\ (g/2)\ \phi \wedge \neg\,(\phi\ \mathbf{r} \to \psi\ \mathbf{r})\\
&\quad = \quad \neg\,(\neg\,\textbf{low-bound}\ (g/2)\ \phi \vee (\phi\ \mathbf{r} \to \psi\ \mathbf{r}))\\
&\quad = \quad \neg\,(\textbf{low-bound}\ (g/2)\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r})
\end{aligned}$$

Besides, as $\mathcal{E}';\Theta \vdash \textbf{non-negl}(g)$, we have $\mathcal{E}';\Theta \vdash \textbf{non-negl}(g^2/4)$ (idem for **det** predicate) by operations on non-negligible real terms. Hence, by the rule G.R-$\tilde{\exists}$, we conclude the following property

$$\mathcal{E}';\Theta \vdash \tilde{\exists}\,(h_g : \textbf{real}).\ \textbf{non-negl}(h_g)\ \tilde{\wedge}\ \textbf{det}(h_g)\ \tilde{\wedge}\ {}_{h_g}[\neg\,(\textbf{low-bound}\ (g/2)\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r})].$$

By characterisation of non-negligibility, we conclude by the rule G.$\tilde{\neg}$:Charac the property

$$\mathcal{E}';\Theta \vdash \tilde{\neg}\,[\textbf{low-bound}\ (g/2)\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r}]$$

Finally, as $\mathcal{E}' \vdash (g : \textbf{real})$ and by the rule G.R-$\tilde{\exists}$, we conclude

$$\mathcal{E};\Theta \vdash \tilde{\exists}\,(g' : \textbf{real}).\ \textbf{non-negl}(g')\ \tilde{\wedge}\ \textbf{det}(g')\tilde{\neg}\,[\textbf{low-bound}\ g'\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r}]$$

Which achieves the proof of soundness of the rule G.LB:Elim by contraposition.

- (G.LB:Intro) We suppose the property $\mathcal{E};\Theta \vdash {}_g[\phi\ r_1\ \ldots\ r_n]$, *i.e.* by definition of the predicate ${}_g[\phi]$ semantics, we suppose the following property

$$\forall\,\eta \in \mathbb{N}^*, \Pr_{\rho\in\Omega}\left[\ [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big). \tag{$*$}$$

We have to prove the following property

$$\forall\,\eta \in \mathbb{N}^*, \Pr_{\rho\in\Omega}\left[\ [\![\textbf{low-bound}\ (g/2)\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![g/2]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

Rules for low-bound predicate

G.LB:Elim
$$\frac{\mathcal{E};\Theta \vdash \tilde{\forall}\, g : \mathbf{real}.\ \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g) \tilde{\to} [\mathbf{low\text{-}bound}\ g\ \phi \to \phi\ \mathbf{r} \to \psi\ \mathbf{r}]}{\mathcal{E};\Theta \vdash [\phi\ \mathbf{r} \to \psi\ \mathbf{r}]}$$

G.LB:Intro
$$\frac{\mathcal{E};\Theta \vdash {}_{g}[\phi\ r_1\ \ldots\ r_n]}{\mathcal{E};\Theta \vdash {}_{g/2}[\mathbf{low\text{-}bound}\ (g/2)\ \phi]}$$

G.LB:Out
$$\frac{\mathcal{E};\Theta \vdash \mathbf{det}(h) \qquad \mathcal{E};\Theta \vdash {}_{g}[\mathbf{low\text{-}bound}\ h\ \phi]}{\mathcal{E};\Theta \vdash {}_{g\cdot h}[(\mathbf{low\text{-}bound}\ h\ \phi) \wedge (\phi\ r_1\ \ldots\ r_n)]}$$

L.LB:Trans
$$\frac{\mathcal{E};\Theta \vdash \mathbf{non\text{-}negl}(g) \qquad \mathcal{E};\Theta \vdash {}_{1}[(\phi\ r_1\ \ldots\ r_n) \to (\psi\ r_1\ \ldots\ r_n)]}{\mathcal{E};\Theta;\varnothing \vdash \mathbf{low\text{-}bound}\ g\ \phi \to \mathbf{low\text{-}bound}\ g\ \psi}$$

G.LB:Trans
$$\frac{\mathcal{E};\Theta \vdash {}_{1}[(\phi\ r_1\ \ldots\ r_n) \to (\psi\ r_1\ \ldots\ r_n)] \qquad \mathcal{E};\Theta \vdash {}_{g}[(\mathbf{low\text{-}bound}\ h\ \phi) \wedge \chi]}{\mathcal{E};\Theta \vdash {}_{g}[(\mathbf{low\text{-}bound}\ h\ \psi) \wedge \chi]}$$

Algebraic rules For the rule L.Open, terms M and $\mathbf{s}$ are defined by $(M, \mathbf{s}) \stackrel{\text{def}}{=} \mathbf{solve\ a}\ (\mathbf{e}_i)_{i=1}^N\ (\mathbf{e}'_i, k_i)_{i=1}^N$.

G.Sel
$$\frac{\mathcal{E};\Theta \vdash \mathbf{det}(k)\ \tilde{\wedge}\ \mathbf{pbound}(k) \qquad \mathcal{E} \vdash \mathbf{r}_s : \mathbf{nat} \to \tau \qquad \mathcal{E} \vdash \mathbf{select}^{(n)}_{\mathrm{rand}} : \mathbf{nat} \to (\mathbf{nat} \to \tau) \to \mathbf{set}_n(\tau) \qquad \mathcal{E};\Theta \vdash [\phi\ (\mathbf{r}_s\ 1)\ \ldots\ (\mathbf{r}_s\ n)] \qquad \mathcal{E};\Theta \vdash [\mathbf{select}^{(n)}_{\mathrm{rand}}\ k\ \mathbf{r}_s \subset \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]}{\mathcal{E};\Theta \vdash [n \leqslant k \to \phi\ (\mathbf{select}^{(n)}_{\mathrm{rand}}\ k\ \mathbf{r}_s)]}$$

L.π:Charac
$$\frac{\mathcal{E};\Theta;\Gamma \vdash M \cdot \mathbf{1} = \mathbf{1} \qquad \mathcal{E};\Theta;\Gamma \vdash \mathbf{prod}_N\ (M \cdot X) - \mathbf{prod}_N\ X = \mathbf{0}}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ M}$$

L.Open
$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{basis}_N\ (\mathbf{e}_i)_{i=1}^N \qquad \mathcal{E};\Theta;\Gamma \vdash \bigwedge_{i=1}^{N} (\mathbf{a} \circledast \mathbf{e}_i = \mathbf{com\text{-}vec}\ ck\ \mathbf{e}'_i\ k_i)}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ ck\ M\ \mathbf{s}}$$

L.Basis
$$\frac{\mathcal{E} \vdash x_1, \ldots x_n : \mathbf{nat} \qquad \mathcal{E} \vdash \mathbf{e}_s : \mathbf{nat} \to \tau \qquad \mathcal{E};\Theta;\Gamma \vdash \bigwedge_{1 \leqslant i < j \leqslant n} x_i \neq x_j}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{basis}_n\ (\mathbf{e}_s\ x_i)_{i=1}^n}$$

L.Eqm:Charac
$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ \pi \qquad \mathcal{E};\Theta;\Gamma \vdash \bigwedge_{i=1}^{N} (\langle \mathbf{x} \mid \mathbf{i} \rangle = \langle \mathbf{x} \mid \pi \cdot \mathbf{i} \rangle)}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{eqm}_N\ \mathbf{x}\ \mathbf{y}}$$

L.Sz
$$\frac{\mathcal{E};\Theta;\Gamma \vdash \phi^{\mathbf{x}_0}_{\mathrm{fresh}}(P) \qquad \mathcal{E};\Theta;\Gamma \vdash P(\mathbf{x}_0) = 0}{\mathcal{E};\Theta;\Gamma \vdash P = \mathbf{0}}$$

L.Wf:Valid
$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ v}{\mathcal{E};\Theta;\Gamma \vdash \exists\, sk.\ \mathbf{wf\_ctxt}\ sk\ c}$$

L.π:Inj
$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ \pi}{\mathcal{E};\Theta;\Gamma \vdash \bigwedge_{i=1}^{N} \bigvee_{j=1}^{N} (M \cdot \mathbf{i} = \mathbf{j})}$$

L.DecList
$$\frac{}{\mathcal{E};\Theta;\Gamma \vdash \langle \mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{x} \mid \mathbf{i} \rangle = \mathbf{dec}_{\mathbb{CS}}\ sk\ \langle \mathbf{x} \mid \mathbf{i} \rangle}$$

L.⊛:CanoVec
$$\frac{}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{x} \circledast \mathbf{i} = \langle \mathbf{x} \mid \mathbf{i} \rangle}$$

L.⊛:Com
$$\frac{}{\mathcal{E};\Theta;\Gamma \vdash (\mathbf{com\text{-}mat}\ ck\ M\ \mathbf{s}) \circledast \mathbf{x} = \mathbf{com\text{-}vec}\ ck\ (M \cdot \mathbf{x})\ \langle \mathbf{s} \mid \mathbf{x} \rangle}$$

L.Shuffle
$$\frac{}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{c}' = \mathbf{shuffle}\ pk\ \mathbf{c}\ \pi\ (\mathbf{r}\ j) \leftrightarrow \bigwedge_{i=1}^{N} (\mathbf{c}' \circledast (\pi \cdot \mathbf{i}) = \mathbf{shuf\text{-}map}\ pk\ (\mathbf{c} \circledast \mathbf{i})\ \langle \mathbf{r}\ j \mid \mathbf{i} \rangle)}$$

Figure 7: New rules

Cryptographic rules: commitment schemes

G.Com:Hide

$$\frac{\mathcal{E};\Theta;\varnothing \vdash_{\text{pptm}} \mathbf{u}, m_1, m_2 \qquad \mathcal{E};\Theta \vdash [\phi^{r,i}_{\text{rand}}(\mathbf{u}, m_1, m_2) \wedge \phi^{ck,n}_{\text{comkey}}(\mathbf{u}, m_1, m_2)]}{\mathcal{E};\Theta \vdash \mathbf{u}, \mathbf{com}\ (ck\ n)\ m_1\ (r\ i) \sim \mathbf{u}, \mathbf{com}\ (ck\ n)\ m_2\ (r\ i)}$$

L.Com:Bind

$$\frac{\mathcal{E};\Theta;\varnothing \vdash_{\text{pptm}} m_1, m_2, r_1, r_2 \qquad \mathcal{E};\Theta;\Gamma \vdash \phi^{ck,n}_{\text{comkey}}(m_1, m_2) \qquad \mathcal{E};\Theta;\Gamma \vdash \mathbf{com}\ (ck\ n)\ m_1\ r_1 = \mathbf{com}\ (ck\ n)\ m_2\ r_2}{\mathcal{E};\Theta;\Gamma \vdash m_1 = m_2}$$

Cryptographic rules: Σ-protocols

L.Σ-P:SpSound

$$\frac{\mathcal{E};\Theta;\Gamma \vdash \bigwedge_{i\in\{1,2\}} \mathbf{zkp\text{-}verif}_{\mathcal{R}}\ x\ \langle \alpha, c_i, z(c_i)\rangle \qquad \mathcal{E};\Theta;\Gamma \vdash c_1 \neq c_2}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{zkp\text{-}rel}_{\mathcal{R}}\ x\ (\mathbf{zkp\text{-}extract}_{\mathcal{R}}\ x\ \mathbf{p}^{(1)}_{\mathcal{R}}(c_1)\ \mathbf{p}^{(2)}_{\mathcal{R}}(c_2))}$$

G.Σ-P:HVZK

$$\frac{\mathcal{E};\Theta;\varnothing \vdash_{\text{pptm}} \mathbf{u}, x, w \qquad \mathcal{E};\Theta \vdash [\phi^{r,i}_{\text{rand}}(\mathbf{u}, x, w)]}{\mathcal{E};\Theta \vdash \mathbf{u}, \mathbf{zkp\text{-}prove}_{\mathcal{R}}\ x\ w\ (r\ i) \sim \mathbf{u}, \mathbf{zkp\text{-}sim}_{\mathcal{R}}\ x\ (r\ i)}$$

Cryptographic rules: *shuffle-friendly* maps

L.Sfm:Correct

$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{wf\_ctxt}\ sk\ c \qquad \mathcal{E};\Theta;\Gamma \vdash \exists\, v.\ c' = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ v}{\mathcal{E};\Theta;\Gamma \vdash \mathbf{dec}_{\mathbb{CS}}\ sk\ c = \mathbf{dec}_{\mathbb{CS}}\ sk\ c'}$$

L.Sfm:Charac

$$\frac{\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ \pi \qquad \mathcal{E};\Theta;\Gamma \vdash \phi^{\mathbf{e}}_{\text{fresh}}(\mathbf{c}, \mathbf{c}', \pi) \qquad \mathcal{E};\Theta;\Gamma \vdash \exists\, v.\ \mathbf{c}' \circledast (\pi \cdot \mathbf{e}) = \mathbf{shuf\text{-}map}\ pk\ (\mathbf{c} \circledast \mathbf{e})\ v}{\mathcal{E}, (\mathbf{x} : \mathbf{msg});\Theta;\Gamma \vdash \exists\, v_{\mathbf{x}}.\ \mathbf{c}' \circledast (\pi \cdot \mathbf{x}) = \mathbf{shuf\text{-}map}\ pk\ (\mathbf{c} \circledast \mathbf{x})\ v_{\mathbf{x}}}$$

G.Sfm:IndCPA

$$\frac{\mathcal{E};\Theta;\varnothing \vdash_{\text{pptm}} \mathbf{u}, c, v \qquad \mathcal{E};\Theta \vdash [\phi^{sk}_{\text{skey}}(\mathbf{u}, c, v)]}{\begin{array}{l}\mathcal{E};\Theta \vdash \mathbf{u}, \mathbf{if\ valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ v\ \mathbf{then\ shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ r \\ \quad \sim\ \mathbf{u}, \mathbf{if\ valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ c\ v\ \mathbf{then\ shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{0}\ (\mathbf{len}\ c))\ r\end{array}}$$

Figure 8: New cryptographic rules

Let $\eta \in \mathbb{N}^*$. By definition of the predicate **low-bound** semantics, we have

$$\Pr_{\rho\in\Omega}\Big[\ [\![\textbf{low-bound}\ (g/2)\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \Big] = \Pr_{\rho\in\Omega}\Big[\ \Pr_{r_i\in[\![\tau_i]\!]^{\eta}_{\mathbb{M}}, i\in[\![1;n]\!]}\Big[\ [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n)\ \Big] \geqslant \mathbb{E}_{\rho'\in\Omega}\big([\![g/2]\!]^{\eta,\rho'}_{\mathbb{M}:\mathcal{E}}\big)\ \Big].$$

Moreover, by semantics of real terms, we have $[\![g/2]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} = \frac{1}{2}[\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}$. Therefore, by linearity of the function $\mathbb{E}_\rho\big(X(\rho)\big)$, we have

$$\mathbb{E}_{\rho\in\Omega}\big([\![g/2]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big) = \frac{1}{2}\mathbb{E}_{\rho\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

We denote by p_ϕ the function defined by

$$p_\phi(\eta,\rho) \stackrel{\text{def}}{=} \Pr_{r_i\in[\![\tau_i]\!]^{\eta}_{\mathbb{M}}, i\in[\![1;n]\!]}\Big[\ [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n)\ \Big].$$

and we denote by e_g the function defined by

$$e_g(\eta) \stackrel{\text{def}}{=} \mathbb{E}_{\rho\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

Consequently, we have to prove the following property

$$\Pr_{\rho\in\Omega}\Big[\ p_\phi(\eta,\rho) \geqslant \frac{1}{2}e_g(\eta)\ \Big] \geqslant \frac{1}{2}e_g(\eta).$$

We have, by definition of $p_\phi(\eta,\rho)$,

$$\Pr_{\rho\in\Omega}\Big[\ [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \Big] = \int_{\rho\in\Omega} p_\phi(\eta,\rho)\,d\rho$$

Hence, the idea is to split the space of random tapes Ω whether or not $p_\phi(\eta,\rho)$ is greater than $\frac{1}{2}e_g(\eta)$. To do so, we denote by Ω_{inf}, respectively Ω_{sup}, the set of random tapes defined by

$$\Omega_{\text{sup}} \stackrel{\text{def}}{=} \big\{\rho\in\Omega\ \mid\ p_\phi(\eta,\rho) \geqslant \tfrac{1}{2}e_g(\eta)\big\}$$
$$\Omega_{\text{inf}} \stackrel{\text{def}}{=} \big\{\rho\in\Omega\ \mid\ p_\phi(\eta,\rho) < \tfrac{1}{2}e_g(\eta)\big\}.$$

As $\Omega = \Omega_{\text{inf}} \sqcup \Omega_{\text{sup}}$ (these two subsets form a partition of the random tape space Ω), we conclude

$$\Pr_{\rho\in\Omega}\Big[\ [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \Big] = \int_{\rho\in\Omega_{\text{inf}}} p_\phi(\eta,\rho)\,d\rho + \int_{\rho\in\Omega_{\text{sup}}} p_\phi(\eta,\rho)\,d\rho$$

Besides, on the set Ω_{inf}, we have by definition of this subset $p_\phi(\eta,\rho) \leqslant \frac{1}{2}e_g(\eta)$. As for the set Ω_{sup}, because $p_\phi(\eta,\rho)$ is a probability, we have $p_\phi(\eta,\rho) \leqslant 1$. Therefore, we have

$$\Pr_{\rho\in\Omega}\Big[\ [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \Big] \leqslant \int_{\rho\in\Omega_{\text{inf}}}\Big(\tfrac{1}{2}e_g(\eta)\Big)\,d\rho + \int_{\rho\in\Omega_{\text{sup}}} d\rho = \frac{1}{2}e_g(\eta)\int_{\rho\in\Omega_{\text{inf}}} d\rho + \int_{\rho\in\Omega_{\text{sup}}} d\rho$$

Now, by property on probabilities, we have

$$\int_{\rho\in\Omega_{\text{inf}}} d\rho \leqslant \Pr_{\rho\in\Omega}\Big[\ p_\phi(\eta,\rho) \leqslant \frac{1}{2}e_g(\eta)\ \Big] \leqslant 1$$

and

$$\int_{\rho\in\Omega_{\text{sup}}} d\rho \leqslant \Pr_{\rho\in\Omega}\Big[\ p_\phi(\eta,\rho) \geqslant \frac{1}{2}e_g(\eta)\ \Big]$$

Moreover, by hypothesis Eq. (∗), we conclude

$$e_g(\eta) \leqslant \Pr_{\rho\in\Omega}\Big[\ [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \Big] \leqslant \frac{1}{2}e_g(\eta) + \Pr_{\rho\in\Omega}\Big[\ p_\phi(\eta,\rho) \geqslant \frac{1}{2}e_g(\eta)\ \Big]$$

Therefore, we have the following probability

$$\forall\, \eta \in \mathbb{N}^*, \mathrm{Pr}_{\rho\in\Omega}\left[\, p_\phi(\eta,\rho) \geqslant \frac{1}{2} e_g(\eta) \,\right] \geqslant \frac{1}{2} e_g(\eta)$$

which achieves the soundness proof of the rule G.LB:Intro.

- (G.LB:Out) We suppose the property $\mathcal{E};\Theta \vdash {}_g[\textbf{low-bound}\ h\ \phi]$ where the parameter h verifies $\mathcal{E};\Theta \vdash \textbf{non-negl}(h)\ \tilde{\wedge}\ \textbf{det}(h)$ By definition of ${}_g[\phi]$ semantics, this leads to the following property

$$\forall\, \eta \in \mathbb{N}^*, \mathrm{Pr}_{\rho\in\Omega}\left[\, [\![\textbf{low-bound}\ h\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big). \tag{$\mathcal{H}$}$$

Let $\eta \in \mathbb{N}^*$ be a security parameter. By definition of $\wedge$ semantics, we have

$$\begin{aligned}\mathrm{Pr}_{\rho\in\Omega}\left[\, [\![(\textbf{low-bound}\ h\ \phi) \wedge (\phi\ r_1\ \ldots\ r_n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] = \mathrm{Pr}_{\rho\in\Omega}\left[\, [\![\textbf{low-bound}\ h\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \\ \cdot\, \mathrm{Pr}_{\rho\in\Omega}\left[\, [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \mid [\![\textbf{low-bound}\ h\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right]\end{aligned}$$

Besides, by definition of **low-bound** semantics, we have

$$\mathrm{Pr}_{r_i\in[\![\tau_i]\!]^\eta_{\mathbb{M}}, i\in[\![1;n]\!]}\left[\, [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n) \,\right] \geqslant \mathbb{E}_{\rho'}\big([\![h]\!]^{\eta,\rho'}_{\mathbb{M}:\mathcal{E}}\big).$$

Moreover, for all $\rho \in \Omega$, we have the following lower bound

$$\mathrm{Pr}_{\rho'\in\Omega}\left[\, [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho'}_{\mathbb{M}:\mathcal{E}} \,\right] \geqslant \mathrm{Pr}_{r_i\in[\![\tau_i]\!]^\eta_{\mathbb{M}}, i\in[\![1;n]\!]}\left[\, [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n) \,\right].$$

Therefore, by the two previous equations, we have the following lower bound

$$\mathrm{Pr}_{\rho\in\Omega}\left[\, [\![\phi\ r_1\ \ldots\ r_n]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \mid [\![\textbf{low-bound}\ h\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![h]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

Consequently, by hypothesis Eq. ($\mathcal{H}$) and by the previous equation, we have

$$\mathrm{Pr}_{\rho\in\Omega}\left[\, [\![(\textbf{low-bound}\ h\ \phi) \wedge (\phi\ r_1\ \ldots\ r_n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big) \cdot \mathbb{E}_{\rho\in\Omega}\big([\![h]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big)$$

As $\mathcal{E};\Theta \vdash \textbf{det}(h)$ holds, we have $\mathbb{E}_{\rho\in\Omega}\big([\![h]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big) = [\![h]\!]^{\eta,}_{\mathbb{M}:\mathcal{E}}$ (here we blend h with its deterministic semantics). Therefore, we conclude the following lower bound by properties on expected value

$$\mathrm{Pr}_{\rho\in\Omega}\left[\, [\![(\textbf{low-bound}\ h\ \phi) \wedge (\phi\ r_1\ \ldots\ r_n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![g \cdot h]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big)$$

which achieve proof of soundness for G.LB:Out.

- (L.LB:Trans) Let $\eta \in \mathbb{N}^*$ be a security parameter and $\rho \in \Omega$ be a random tape. We suppose $[\![\textbf{low-bound}\ g\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}$. By definition of **low-bound** semantics, we have the following inequality

$$\mathrm{Pr}_{r_i\in[\![\tau_i]\!]^\eta_{\mathbb{M}}, i\in[\![1;n]\!]}\left[\, [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n) \,\right] \geqslant \mathbb{E}_{\rho'\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

However, by hypothesis $\mathcal{E};\Theta \vdash {}_1[(\phi\ r_1\ \ldots\ r_n) \to (\psi\ r_1\ \ldots\ r_n)]$, we conclude the following inequality

$$\mathrm{Pr}_{r_i\in[\![\tau_i]\!]^\eta_{\mathbb{M}}, i\in[\![1;n]\!]}\left[\, [\![\phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n) \,\right] \leqslant \mathrm{Pr}_{r_i\in[\![\tau_i]\!]^\eta_{\mathbb{M}}, i\in[\![1;n]\!]}\left[\, [\![\psi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n) \,\right].$$

Therefore, the two previous inequalities leads to the following property

$$\mathrm{Pr}_{r_i\in[\![\tau_i]\!]^\eta_{\mathbb{M}}, i\in[\![1;n]\!]}\left[\, [\![\psi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}(r_1,\ldots,r_n) \,\right] \geqslant \mathbb{E}_{\rho'\in\Omega}\big([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

Said otherwise, for all security parameter $\eta \in \mathbb{N}^*$ and for all random tape $\rho \in \Omega$, we conclude $[\![\textbf{low-bound}\ g\ \psi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}$, and achieve the proof.

- (G.LB:Trans) Let h : **real** with **non-negl**(h) be two non-negligible parameter. Let $\psi : \tau_1 \to \cdots \to \tau_n \to$ **bool** be a formula with n parameters. Let $\chi : \tau \to$ **bool** be a formula. By definition of ${}_g[\phi]$ semantics, we have to prove

$$\forall\, \eta \in \mathbb{N}^*, \Pr_{\rho\in\Omega}\left[\, [\![(\textbf{low-bound}\ h\ \psi) \wedge \chi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \geqslant \mathbb{E}_{\rho\in\Omega}\big([\![h]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

Let $\eta \in \mathbb{N}^*$ be a security parameter. Because we have as hypothesis $\mathcal{E};\Theta \vdash {}_1[(\phi\ r_1\ \ldots\ r_n) \to (\psi\ r_1\ \ldots\ r_n)]$, we conclude by the rule L.LB:Trans the following property

$$\mathcal{E};\Theta;\varnothing \vdash \textbf{low-bound}\ h\ \phi \wedge \chi \to \textbf{low-bound}\ h\ \psi \wedge \chi.$$

Therefore, we have the following inequality

$$\Pr_{\rho\in\Omega}\left[\, [\![(\textbf{low-bound}\ h\ \phi) \wedge \chi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \leqslant \Pr_{\rho\in\Omega}\left[\, [\![(\textbf{low-bound}\ h\ \psi) \wedge \chi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right].$$

Then, by the hypothesis $\mathcal{E};\Theta \vdash {}_g[(\textbf{low-bound}\ h\ \phi) \wedge \chi]$, we conclude the property we want.

B.2 Soundness of algebraic rules

- (G.Sel) Let k : **nat** be a polynomial bounded and deterministic natural term. As k is deterministic, that is the semantics of k does not depend on the random tape ρ (*i.e.* for all random tapes $\rho, \rho' \in \Omega$ and for all security parameter $\eta \in \mathbb{N}^*$, we have $[\![k]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} = [\![k]\!]^{\eta,\rho'}_{\mathbb{M}:\mathcal{E}}$). Therefore, we denote by k the function $k : \eta \longmapsto [\![k]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}$ and because **pbound**(k) holds, the related function k is polynomial bounded. Let $\mathbf{r}_s$: **nat** $\to \tau$ be a random source term of random terms of type τ *uniformly distributed*. Let $n \in \mathbb{N}^*$ be a natural number and $\phi : \tau \to \cdots \to \tau \to$ **bool** be a formula of n parameters of the same type τ. Let $\textbf{select}^{(n)}_{\text{rand}}$: **nat** $\to$ (**nat** $\to \tau$) $\to$ **set**$_n(\tau)$ be an adversarial selection function of n distinct terms of type τ given by a random source term. Let $\eta \in \mathbb{N}^*$ be a security parameter. We suppose that we are in the case where $[\![k]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \leqslant n$. By definition of the type **set**$_n(\tau)$, we have $\text{Card}([\![\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}) = n$. Then, by hypothesis $\mathcal{E};\Theta \vdash [\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]$, we conclude, without loss of generality, the existency of n distinct natural terms $t_1, \ldots, t_n$: **nat**, such that $1 \leqslant t_1 < \cdots < t_n \leqslant k$ and $\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s = \{\mathbf{r}_s\ t_i\}_{i=1}^n$. Therefore, we have

$$\Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] = \Pr_{\rho\in\Omega}\left[\, \exists\, 1 \leqslant j_1 < \cdots < j_n \leqslant k(\eta), [\![\neg\ \phi\ (\mathbf{r}_s\ j_1)\ \ldots\ (\mathbf{r}_s\ j_n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right]$$

By property on probabilities, we have the following upper bound

$$\Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \leqslant \sum_{\{j_i\}_{i=1}^n\ \subseteq\ [\![1;k(\eta)]\!]} \Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\mathbf{r}_s\ j_1)\ \ldots\ (\mathbf{r}_s\ j_n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right].$$

But as $\mathbf{r}_s$ is a random source term of *uniformly distributed* random terms of type τ, we have

$$\forall\, \{j_i\}_{i=1}^n\ \subseteq\ [\![1;k(\eta)]\!],\ \Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\mathbf{r}_s\ j_1)\ \ldots\ (\mathbf{r}_s\ j_n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] = \Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\mathbf{r}_s\ 1)\ \ldots\ (\mathbf{r}_s\ n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right].$$

Besides, we have

$$\text{Card}(\{j_i\}_{i=1}^n\ \subseteq\ [\![1;k(\eta)]\!]) = \binom{k(\eta)}{n} \leqslant k(\eta)^n.$$

Therefore, we conclude the following upper bound

$$\Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right] \leqslant \underbrace{k(\eta)^n}_{\text{polynomial in } \eta} \cdot \underbrace{\Pr_{\rho\in\Omega}\left[\, [\![\neg\ \phi\ (\mathbf{r}_s\ 1)\ \ldots\ (\mathbf{r}_s\ n)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \,\right]}_{\text{negligible in } \eta}.$$

By hypothesis $\mathcal{E};\Theta \vdash [\phi\ (\mathbf{r}_s\ 1)\ \ldots\ (\mathbf{r}_s\ n)]$, we conclude the property we want, *i.e.* we obtain the following property

$$\mathcal{E};\Theta \vdash [n \leqslant k \to \phi\ (\textbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s)].$$

- (L.π:Charac) This rule is a model of the following proposition:

  Proposition B.1 (Characterisation of permutation matrix)**.** Let $M \in \mathbb{Z}_{q_\eta}^{N\times N}$ be a matrix. Let $\mathbf{e} \xleftarrow{\$} \mathbb{Z}_{q_\eta}^{N}$ be a vector of N independent variables and chosen uniformly at random. We suppose that the two following equations, denoted by (i) and (ii) hold, for M and $\mathbf{e}$.

  $$(i) \quad M \cdot \mathbf{1} = \mathbf{1} \quad \text{and} \quad (ii) \quad \prod_{i=1}^{N} \big(M \cdot \mathbf{e}\big)_i = \prod_{i=1}^{N} e_i.$$

  Then we conclude that M is a permutation matrix with probability at least equal to $1 - \frac{N}{q_\eta^N}$.

  A proof of this proposition can be found in [19].

- (L.Open) Soundness of this rule come quite straightforwardly from the following lemma:

  Lemma B.1. Let $\mathbf{a} = (a_i)_{i=1}^{N} \in \mathbb{G}_{q_\eta}^{N}$ be a vector. Suppose there exists a set $\mathcal{W} = \{\mathbf{e}_i\}_{i=1}^{N}$ of N linearly independent vectors of $\mathbb{Z}_{q_\eta}^{N}$ such that

  $$\forall\, i \in [\![1; N]\!], \exists\, \mathbf{e}'_i \in \mathbb{Z}_{q_\eta}^{N}, \exists\, k_i \in \mathbb{Z}_{q_\eta},\ \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}\left(ck, \mathbf{e}'_{\mathbf{i}}\,;\, k_i\right) = \mathbf{a} \circledast \mathbf{e}_{\mathbf{i}}.$$

  Then $\mathbf{a}$ is a commit message to a matrix $M_{\mathcal{W}} \in \mathbb{Z}_{q_\eta}^{N\times N}$ using the vector of random values $\mathbf{s}_{\mathcal{W}} \in \mathbb{Z}_{q_\eta}^{N}$, *i.e.* we have
  $\mathbf{a} = \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N\times N}}\left(ck, M_{\mathcal{W}}\,;\, \mathbf{s}_{\mathcal{W}}\right)$. Besides, these opening $(M_{\mathcal{W}}, \mathbf{s}_{\mathcal{W}})$ can be obtained in polynomial time.

  Proof. Let $\mathbf{a} = (a_i)_{i=1}^{N} \in \mathbb{G}_{q_\eta}^{N}$ be a vector of values in the group $\mathbb{G}_{q_\eta}$. Let $\mathcal{W} = \{\mathbf{e}_i\}_{i=1}^{N}$ a set of N linearly independent vectors of $\mathbb{Z}_{q_\eta}^{N}$ such that

  $$\forall\, i \in [\![1; N]\!], \exists\, \mathbf{e}'_i \in \mathbb{Z}_{q_\eta}^{N}, \exists\, k_i \in \mathbb{Z}_{q_\eta},\ \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}\left(ck, \mathbf{e}'_i\,;\, k_i\right) = \mathbf{a} \circledast \mathbf{e}_i. \tag{$*$}$$

  As the vectors of the set $\mathcal{W} = \{\mathbf{e}_i\}_{i=1}^{N}$ are linearly independent, and because $\dim(\mathbb{Z}_{q_\eta}^{N}) = N$, the family $\mathcal{B}_{\mathcal{W}} = \big(\mathbf{e}_1, \ldots, \mathbf{e}_N\big)$ is a basis of $\mathbb{Z}_{q_\eta}^{N}$. Hence, for all $j \in [\![1; N]\!]$, there exists a set of scalar values $\{\lambda_i^{(j)}\}_{i=1}^{N} \in \mathbb{Z}_{q_\eta}^{N}$ such that $\sum_{i=1}^{N} \lambda_i^{(j)} \mathbf{e}_i = \mathbf{u}_j$ where $\mathbf{u}_j$ is the j-th standard vector of $\mathbb{Z}_{q_\eta}^{N}$. In fact, such set of scalar values can be obtain in polynomial time by Gaussian elimination. Let $j \in [\![1; N]\!]$. By basic properties on $\circledast$, we have

  $$a_j = \mathbf{a} \circledast \left(\sum_{i=1}^{N} \lambda_i^{(j)} \mathbf{e}_i\right) = \prod_{i=1}^{N} \left(\mathbf{a} \circledast \mathbf{e}_i\right)^{\lambda_i^{(j)}}.$$

  By the equation Eq. ($*$), we have, for all $i \in [\![1; N]\!]$, $\mathbf{a} \circledast \mathbf{e}_i = \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}\left(ck, \mathbf{e}'_i\,;\, k_i\right)$. Thus,

  $$a_j = \prod_{i=1}^{N} \left(\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}\left(ck, \mathbf{e}'_i\,;\, k_i\right)\right)^{\lambda_i^{(j)}}$$

  By definition of the commitment algorithm $\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}$, we have $\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}\left(ck, \mathbf{e}'_i\,;\, k_i\right) = g^{k_i} \prod_{l=1}^{N} g_l^{(\mathbf{e}'_i)_l}$. Consequently,

  $$a_j = \prod_{i=1}^{N} \left(g^{\lambda_i^{(j)} k_i} \prod_{l=1}^{N} g_l^{\lambda_i^{(j)} (\mathbf{e}'_i)_l}\right) = g^{\sum_{i=1}^{N} \lambda_i^{(j)} k_i} \prod_{l=1}^{N} g_l^{\sum_{i=1}^{N} \lambda_i^{(j)} (\mathbf{e}'_i)_l}$$

  Finally, we have

  $$\forall\, j \in [\![1; N]\!], a_j = \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}\left(ck, \sum_{i=1}^{N} \lambda_i^{(j)} \mathbf{e}'_i\ ;\ \sum_{i=1}^{N} \lambda_i^{(j)} k_i\right).$$

Consequently, we conclude that $\mathbf{a}$ is indeed a commit message produced by the commitment algorithm $\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N\times N}}$, *i.e.* $\mathbf{a} = \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N\times N}}(ck, M_{\mathcal{W}}\,;\,\mathbf{s}_{\mathcal{W}})$ where $M_{\mathcal{W}} \in \mathbb{Z}_{q_\eta}^{N\times N}$ and $\mathbf{s}_{\mathcal{W}} \in \mathbb{Z}_{q_\eta}^{N}$ are defined as following.

$$\boxed{M_{\mathcal{W}} = \left(\sum_{j=1}^{N} \lambda_j^{(l)} \mathbf{e}'_j\right)_{l=1}^{N} \qquad \text{and} \qquad \mathbf{s}_{\mathcal{W}} = \left(\sum_{j=1}^{N} \lambda_j^{(l)} k_j\right)_{l=1}^{N}}$$

□

- (L.Basis) Let $n \in \mathbb{N}^*$ be a non-null natural number. Let $(\mathbf{e}_i)_{i=1}^{n-1}$ be a free family of vector in $\mathbb{Z}_{q_\eta}^n$. Let $\mathbb{H}$ be the linear span of vectors set $(\mathbf{e}_i)_{i=1}^{n-1}$. Hence, $\mathbb{H}$ defines a hyperplane of $\mathbb{Z}_{q_\eta}^n$. Therefore, the probability to choose a new vector $\mathbf{e}$ uniformly and independently from vectors family $(\mathbf{e}_i)_{i=1}^{n-1}$ such that $\mathbf{e} \in \mathbb{H}$ is at most equal to $\frac{1}{q_\eta}$:

$$\Pr_{\mathbf{e}\xleftarrow{\$}\mathbb{Z}_{q_\eta}^n}\left[\,\mathbf{e} \in \mathbb{H}\,\right] \leqslant \frac{1}{q_\eta}.$$

  Which achieve proof of soundness of the rule L.Basis.

- (L.π:Charac) This rule is a model of the following lemma:

  Lemma B.2 (Schwartz-Zippel)**.** Let $f_d \in \mathbb{Z}_{q_\eta}[X_1, \ldots, X_N]$ be a non-zero multivariate polynomial of total degree $d \in \mathbb{N}$ over $\mathbb{Z}_{q_\eta}$. Let $\mathbf{e} \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N$ be a vector chosen uniformly at random in the vector space $\mathbb{Z}_{q_\eta}^N$. Then $\Pr_{\mathbf{e}\in\mathbb{Z}_{q_\eta}^N}\left[\,f_d(\mathbf{e}) = 0\,\right] \leqslant \frac{d}{q_\eta^N}$.

  A proof of this lemma can be found in [20] and [17].

- Soundness of rules L.Eqm:Charac, L.Wf:Valid, L.π:Inj, L.DecList, L.$\circledast$:CanoVec and L.Shuffle are trivial by definition of involved functions or predicates.

- (L.$\circledast$:Com) Soundness of this rule come from the following proposition:

  Proposition B.2. For $ck = (g, \mathbf{g}) \leftarrow \mathrm{Gen}(1^\eta, N)$ be a commitment key, for all matrix $M \in \mathbb{Z}_{q_\eta}^{N\times N}$ and for all vectors $\mathbf{x}, \mathbf{s} \in \mathbb{Z}_{q_\eta}^N$, we have the following identity.

$$\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N\times N}}(ck, M\,;\,\mathbf{s}) \circledast \mathbf{x} = \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck, M\cdot x\,;\,\langle \mathbf{s} \mid \mathbf{x}\rangle)$$

  Proof. Let $ck = (g, g_1, \ldots, g_N) \leftarrow \mathrm{Gen}(1^\eta, N)$ be a commitment key. Let $M \in \mathbb{Z}_{q_\eta}^{N\times N}$ be a matrix and let $\mathbf{x}, \mathbf{s} \in \mathbb{Z}_{q_\eta}^N$ be two vectors. Then, by definitions of both commitment schemes and of operator $\circledast$, we have

$$\begin{aligned}
\mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N\times N}}(ck, M\,;\,\mathbf{s}) \circledast \mathbf{x} &= \prod_{i=1}^{N} g^{s_i x_i} \prod_{j=1}^{N} g_j^{m_{j,i} x_i} \\
&= g^{\sum_{i=1}^{N} s_i x_i} \prod_{j=1}^{N} g_j^{\sum_{i=1}^{N} m_{j,i} x_i} \\
&= \mathrm{Com}_{\mathbb{Z}_{q_\eta}^{N}}(ck, M\cdot\mathbf{x}\,;\,\langle \mathbf{s} \mid \mathbf{x}\rangle).
\end{aligned}$$

□

B.3 Soundness of cryptographic rules

Using semantics of functions appearing in cryptographic rules and associated with corresponding cryptographic assumptions, soundness of rules G.Com:Hide, L.Com:Bind, L.Σ-P:SpSound, G.Σ-P:HVZK and G.Sfm:IndCPA are proved as other cryptographic rules proofs found in [3]. As soundness of rule L.Sfm:Correct is strongly dependent from definition of *shuffle-friendly* maps, this rule has to be proved as soon as such map is defined. It remains to prove soundness of the rule L.Sfm:Charac of characterisation for *shuffle-friendly* maps. Actually, soundness of this rule come from the following lemma we will prove next.

Lemma B.3 (Characterisation of correct shuffle)**.** Let ϕ_{pk} be a *shuffle-friendly* map for a cryptosystem $\mathbb{CS}$. Let $(c_1, \ldots, c_N) \in \mathcal{C}_{\mathbb{CS}}^N$ and $(c'_1, \ldots, c'_N) \in \mathcal{C}_{\mathbb{CS}}^N$ be two lists of ciphertexts. Let $\pi \in \mathfrak{S}_N$ be a permutation of length N. Let $pk \in \mathcal{PK}_{\mathbb{CS}}$ be a public-key for the cryptosystem $\mathbb{CS}$. We denote by $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi} \subseteq \mathbb{Z}_{q_\eta}^N$ the following set

$$\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi} = \left\{ \ \mathbf{e} \in \mathbb{Z}_{q_\eta}^N \ \middle| \ \exists\, v \in \mathbb{Z}_{q_\eta}, \mathbf{c}' \circledast \left(M_\pi \cdot \mathbf{e}\right) = \phi_{pk}(\mathbf{c} \circledast \mathbf{e}\,;\, v) \ \right\}$$

Then, we have an equivalence between the following properties.

(i) There exists a vector of random values $\mathbf{r} = (r_i)_{i=1}^N \in \mathbb{Z}_{q_\eta}^N$ such that we have:

$$\forall\, i \in [\![1; N]\!], c'_{\pi(i)} = \phi_{pk}(c_i\,;\, r_i).$$

(ii) $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi} = \mathbb{Z}_{q_\eta}^N$.

(iii) $\mathrm{Card}(\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}) > q_\eta^{N-1}$.

(iv) $\Pr_{\mathbf{e} \in \mathbb{Z}_{q_\eta}^N}\left[\ \mathbf{e} \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}\ \right] > \dfrac{1}{q_\eta}$.

Proof. Let ϕ_{pk} be a *shuffle-friendly* map for a cryptosystem $\mathbb{CS}$. Let $(c_1, \ldots, c_N) \in \mathcal{C}_{\mathbb{CS}}^N$ and $(c'_1, \ldots, c'_N) \in \mathcal{C}_{\mathbb{CS}}^N$ be two lists of ciphertexts. Let $\pi \in \mathfrak{S}_N$ be a permutation of length N. Let $pk \in \mathcal{PK}_{\mathbb{CS}}$ be a public-key for the cryptosystem $\mathbb{CS}$.

- $\underline{(i) \implies (ii)}$ *Suppose there exists a vector of random values* $\mathbf{r} = (r_i)_{i=1}^N \in \mathbb{Z}_{q_\eta}^N$ *such that:* $\forall\, i \in [\![1; N]\!], c'_{\pi(i)} = \phi_{pk}(c_i\,;\, r_i)$. We want to prove the following inclusion: $\mathbb{Z}_{q_\eta}^N \subseteq \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$. Let $\mathbf{e} \in \mathbb{Z}_{q_\eta}^N$ be a vector. Let $M_\pi \in \mathbb{Z}_{q_\eta}^{N \times N}$ be the permutation matrix representing the permutation π. We set $\mathbf{e}' = M_\pi \cdot \mathbf{e}$. By definition of $\mathbf{e}'$, we have, for all $i \in [\![1; N]\!]$,

$$e'_i = \left(M_\pi \cdot \mathbf{e}\right)_i = \sum_{j=1}^N m_{i,j}^{(\pi)} e_j = \sum_{j=1}^N \delta_{i\pi(j)} e_j = e_{\pi^{-1}(i)}.$$

Hence, we have

$$\begin{aligned}
\mathbf{c}' \circledast \mathbf{e}' &= \prod_{i=1}^N (c'_i)^{e_{\pi^{-1}(i)}} = \prod_{i=1}^N \left(\phi_{pk}(c_{\pi^{-1}(i)}\,;\, r_{\pi^{-1}(i)})\right)^{e_{\pi^{-1}(i)}} \\
&\quad \text{(by the hypothesis } (i)\text{)} \\
&= \phi_{pk}\left(pk, \prod_{i=1}^N c_{\pi^{-1}(i)}^{e_{\pi^{-1}(i)}}\ ;\ \sum_{i=1}^N e_{\pi^{-1}(i)} r_{\pi^{-1}(i)}\right) \\
&\quad \text{(because } \phi_{pk} \text{ is a homomorphism)} \\
&= \phi_{pk}(\mathbf{c} \circledast \mathbf{e}\,;\, \langle \mathbf{e} \mid \mathbf{r} \rangle).
\end{aligned}$$

Thus, we have $\mathbf{e} \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$, *i.e.* we have proved (ii).

- $\underline{(ii) \implies (i)}$ Actually, we will proceed by contraposition. Hence, we suppose the existence of $i_0 \in [\![1; N]\!]$ such that we have the following property

$$\forall\, v \in \mathbb{Z}_{q_\eta}, c'_{\pi(i_0)} \neq \phi_{pk}(c_{i_0}\,;\, v).$$

We will show that $\mathbf{u}_{i_0} \notin \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$. Let $v \in \mathbb{Z}_{q_\eta}$. Hence, we have.

$$\begin{aligned}
\mathbf{c}' \circledast \big(M_\pi \cdot \mathbf{u}_{i_0}\big) &= c'_{\pi(i_0)} \\
&\neq \phi_{pk}(c_{i_0}\,;\, v) && \text{(by definition of } \neg\,(i)) \\
&= \phi_{pk}(\mathbf{c} \circledast \mathbf{u}_{i_0}\,;\, v)
\end{aligned}$$

Consequently, we have $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi} \subsetneq \mathbb{Z}_{q_\eta}^N$.

- $\underline{(ii) \iff (iii)}$ In fact, we will prove that $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$ is a subgroup of $\big(\mathbb{Z}_{q_\eta}^N, +\big)$. Let $\mathbf{e}_1, \mathbf{e}_2 \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$. By definition of $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$, there exists $v_1, v_2 \in \mathbb{Z}_{q_\eta}$ such that

$$\begin{cases} \mathbf{c}' \circledast \big(M_\pi \cdot \mathbf{e}_1\big) = \phi_{pk}(\mathbf{c} \circledast \mathbf{e}_1\,;\, v_1) \\ \mathbf{c}' \circledast \big(M_\pi \cdot \mathbf{e}_2\big) = \phi_{pk}(\mathbf{c} \circledast \mathbf{e}_2\,;\, v_2) \end{cases}$$

Then, we have

$$\begin{aligned}
\mathbf{c}' \circledast \big(M_\pi \cdot (\mathbf{e}_1 - \mathbf{e}_2)\big) &= \mathbf{c}' \circledast \big(M_\pi \cdot \mathbf{e}_1\big) \cdot \Big(\mathbf{c}' \circledast \big(M_\pi \cdot \mathbf{e}_2\big)\Big)^{-1} \\
&= \phi_{pk}(\mathbf{c} \circledast \mathbf{e}_1\,;\, v_1) \cdot \Big(\phi_{pk}(\mathbf{c} \circledast \mathbf{e}_2\,;\, v_2)\Big)^{-1} \\
&\text{(because } \mathbf{e}_1 \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi} \text{ and } \mathbf{e}_2 \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}) \\
&= \phi_{pk}(\mathbf{c} \circledast (\mathbf{e}_1 - \mathbf{e}_2)\,;\, v_1 - v_2) \\
&\text{(by a basic property of } \phi_{pk})
\end{aligned}$$

Consequently, we have $\mathbf{e}_1 - \mathbf{e}_2 \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$. Thus, $\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}$ is a subgroup of $\big(\mathbb{Z}_{q_\eta}^N, +\big)$. However, by the Lagrange's theorem, the cardinal $\mathrm{Card}(\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi})$ divides the cardinal $\mathrm{Card}(\mathbb{Z}_{q_\eta}^N) = q_\eta^N$. Therefore, we have $(ii) \iff (iii)$.

- $\underline{(iii) \iff (iv)}$ As the vector $\mathbf{e} \xleftarrow{\$} \mathbb{Z}_{q_\eta}^N$ is *chosen uniformly at random*, we have

$$\mathrm{Pr}_{\mathbf{e} \in \mathbb{Z}_{q_\eta}^N}\Big[\ \mathbf{e} \in \mathbb{H}_{\mathbf{c},\mathbf{c}',\pi}\ \Big] = \frac{\mathrm{Card}(\mathbb{H}_{\mathbf{c},\mathbf{c}',\pi})}{q_\eta^N}.$$

Consequently, we have $(iii) \iff (iv)$.

□

C Rewinding algorithms

The procedure of witness extraction for the Σ-protocol $\Sigma_{\mathcal{R}}$ is given by Algorithm 1.

The following procedure given in Algorithm 2 which defines an adversarial selection function $\mathbf{select}_{\mathrm{rand}}^{(n)} : \mathbf{nat} \to (\mathbf{nat} \to \tau) \to \mathbf{set}_n(\tau)$

Algorithm 1: Witness extraction procedure using the rewinding technique

Input : A security parameter $\eta \in \mathbb{N}^*$. An adversary $\mathcal{A}$. A Σ-protocol $\Sigma_{\mathcal{R}} = (\mathcal{S}, \mathcal{P}, \mathcal{V})$ for a computable binary relation $\mathcal{R}$. An extractor $\mathcal{E}_{\mathcal{R}}$ for $\Sigma_{\mathcal{R}}$. A statement $x_\eta \in \mathcal{L}_{\mathcal{R}}$ of bit-size polynomial in the security parameter η, *i.e.* $|x_\eta| = \eta^{O(1)}$.

Output: A witness $w \in \mathcal{W}_{\mathcal{R}}$ such that $(x_\eta, w) \in \mathcal{R}$.

let ***extract-sigp***$_{\mathcal{R}}$ x_η **=**

  The adversary $\mathcal{A}$ begins by computing some commit message for the statement x_η which updates her state and sends it to the verifier: $(\mathtt{st}_{\mathcal{A}}^{(1)}, \alpha) \leftarrow \mathcal{A}(x_\eta)$;

  repeat

    The verifier $\mathcal{V}$ chooses a first challenge $c_1 \leftarrow \mathcal{V}(x_\eta, \alpha)$;

    $\mathcal{A}$ produces a response for this challenge, which also updates her state: $(\mathtt{st}_{\mathcal{A}}^{(2)}, z_1(c_1)) \leftarrow \mathcal{A}(x, \alpha, c_1 \ ; \ \mathtt{st}_{\mathcal{A}}^{(1)})$;

    Then, we *rewind* $\mathcal{A}$ to her previous state $\mathtt{st}_{\mathcal{A}}^{(1)}$;

    One more time, $\mathcal{V}$ chooses a second challenge $c_2 \leftarrow \mathcal{V}(x_\eta, \alpha)$ and $\mathcal{A}$ produces another response for this challenge: $(\mathtt{st}_{\mathcal{A}}^{(2')}, z_2(c_2)) \leftarrow \mathcal{A}(x_\eta, \alpha, c_2 \ ; \ \mathtt{st}_{\mathcal{A}}^{(1)})$;

    Finally, the verifier $\mathcal{V}$ check whether or not the two produced proofs are valid $b_i \leftarrow \mathcal{V}(x_\eta, \langle \alpha, c_i, z_i(c_i) \rangle)$;

  until *both booleans b_1 and b_2 are true ($b_1 = b_2 = 1$) and the challenges are different ($c_1 \neq c_2$).*;

  Finally, at this point, we can finally extract the witness from the two proof transcripts $\mathbf{p}_{\mathcal{R}}^{(i)}(c_i) \stackrel{\text{def}}{=} \langle \alpha, c_i, z_i(c_i) \rangle$:

  return $w \leftarrow \mathcal{E}_{\mathcal{R}}(x, \mathbf{p}_{\mathcal{R}}^{(1)}(c_1), \mathbf{p}_{\mathcal{R}}^{(2)}(c_2))$

Algorithm 2: Adversarial selection function for rewinding

Input : A natural number $k \in \mathbb{N}^*$ and a source of *uniformly distributed and independent* random values $\mathbf{r}_s : \mathbb{N}^* \longrightarrow X$. *(implicit inputs)* A natural number $n \in \mathbb{N}^*$ such that $n \leqslant k$ and a formula $\phi_{\eta,\rho} : X \longrightarrow \{0, 1\}$ evaluable in polynomial time.

Output: n random values $(\mathbf{r}_s(i_j))_{j=1}^{n} \subseteq X^n$ with $1 \leqslant i_1 < \ldots < i_n \leqslant k$.

let ***select***$_{rand}^{(n)}$ k $\mathbf{r}_s$ **=**

  $t \leftarrow 1$; $l \leftarrow 1$; $\mathbb{L} \leftarrow []$;

  while $(l \leqslant n \ \wedge \ t \leqslant k)$ **do**

    $i_l \stackrel{\$}{\leftarrow} [\![1; k]\!] \setminus \{i_j\}_{j=1}^{l-1}$;

    if $\phi_{\eta,\rho}(\mathbf{r}_s(i_l))$ **then**

      $\mathbb{L} \leftarrow \mathbf{r}_s(i_l) :: \mathbb{L}$;

      $l \leftarrow l + 1$;

    $t \leftarrow t + 1$;

  end

  return $\mathbb{L}$

D Full version of security properties proof

D.1 Proof of permutation secrecy

By unfolding the definition of the mix predicate $\mathbf{mix}_{\phi_{pk}}$, one has to prove the following indistinguishability

$$\mathcal{E};\varnothing \vdash \mathbf{a}_\pi, \mathbf{p}_\pi(\pi), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi, \mathbf{p}_\phi(\pi) \sim \mathbf{a}_{\mathrm{id}}, \mathbf{p}_\pi(\mathrm{id}), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}, \mathbf{p}_\phi(\mathrm{id})$$

where $\mathbf{a}_\pi \overset{\text{def}}{=} \mathbf{com\text{-}mat}\ (ck\ n)\ \pi\ (\mathbf{s}\ i_1)$, $\mathbf{p}_\pi(\pi) \overset{\text{def}}{=} \mathbf{zkp\text{-}prove}_\pi\ (ck\ n, \mathbf{a}_\pi, \mathbf{e}_\pi)\ w_\pi\ (r_\pi\ i_3)$, $\mathbf{c}'_\pi \overset{\text{def}}{=} \mathbf{shuffle}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ \pi\ (\mathbf{r}\ i_4)$, and $\mathbf{p}_\phi(\pi) \overset{\text{def}}{=} \mathbf{zkp\text{-}prove}_\phi\ (ck\ n, \mathbf{a}_\pi, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{c}'_\pi, \mathbf{e}_\phi)\ w_\phi\ (r_\phi\ i_6)$

1. By the rule G.Σ-P:HVZK applied to the relation of correct shuffle $\mathcal{R}^{\mathrm{shuffle}}_{\phi_{pk}}([\![\mathbf{e}_\phi]\!]^{\eta\,;\,\rho}_{\mathbb{M}\,:\,\mathcal{E}})$, we have to prove

$$\begin{aligned}\mathcal{E};\varnothing \vdash \mathbf{a}_\pi, \mathbf{p}_\pi(\pi), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi,&\\ \mathbf{zkp\text{-}sim}_\phi\ (ck\ n, \mathbf{a}_\pi, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{c}'_\pi, \mathbf{e}_\phi)\ (r_\phi\ i_6)&\\ \sim \mathbf{a}_{\mathrm{id}}, \mathbf{p}_\pi(\mathrm{id}), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}},&\\ \mathbf{zkp\text{-}sim}_\phi\ (ck\ n, \mathbf{a}_{\mathrm{id}}, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{c}'_{\mathrm{id}}, \mathbf{e}_\phi)\ (r_\phi\ i_6)&\end{aligned}$$

2. By the rule G.Σ-P:HVZK applied to the relation of correct shuffle $\mathcal{R}^{\mathrm{com}}([\![\mathbf{e}_\pi]\!]^{\eta\,;\,\rho}_{\mathbb{M}\,:\,\mathcal{E}})$, we have to prove

$$\begin{aligned}\mathcal{E};\varnothing \vdash \mathbf{a}_\pi, \mathbf{zkp\text{-}sim}_\pi\ (ck\ n, \mathbf{a}_\pi, \mathbf{e}_\pi)\ (r_\pi\ i_3), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}&\\ \mathbf{c}'_\pi, \mathbf{zkp\text{-}sim}_\phi\ (ck\ n, \mathbf{a}_\pi, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{c}'_\pi, \mathbf{e}_\phi)\ (r_\phi\ i_6)&\\ \sim \mathbf{a}_{\mathrm{id}}, \mathbf{zkp\text{-}sim}_\pi\ (ck\ n, \mathbf{a}_{\mathrm{id}}, \mathbf{e}_\pi)\ (r_\pi\ i_3), \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}&\\ \mathbf{c}'_{\mathrm{id}}, \mathbf{zkp\text{-}sim}_\phi\ (ck\ n, \mathbf{a}_{\mathrm{id}}, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{c}'_{\mathrm{id}}, \mathbf{e}_\phi)\ (r_\phi\ i_6)&\end{aligned}$$

3. By the function application rule G.∼:FA and by the elimination of duplicates rule G.∼:Dup, we have to prove

$$\begin{aligned}\mathcal{E};\varnothing \vdash \mathbf{a}_\pi, ck\ n, \mathbf{e}_\pi, r_\pi\ i_3, \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{e}_\phi, r_\phi\ i_6&\\ \sim \mathbf{a}_{\mathrm{id}}, ck\ n, \mathbf{e}_\pi, r_\pi\ i_3, \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}, (\mathtt{pk}_{\mathbb{CS}}\ sk), \mathbf{c}, \mathbf{e}_\phi, r_\phi\ i_6&\end{aligned}$$

4. By eliminate all terms computed by the adversary $\mathbf{e}_\pi$, $\mathbf{e}_\phi$ and $\mathbf{c}$ with the function application rule G.∼:FA and by the freshness rule G.∼:Fresh, we have to prove

$$\mathcal{E};\varnothing \vdash \mathbf{a}_\pi, \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi \sim \mathbf{a}_{\mathrm{id}}, \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}\ \mathbf{else}$$

5. Next, the *hiding* rule G.Com:Hide for the commitment scheme to matrix $\mathbb{KS}(\mathbb{Z}_{q_\eta}^{N\times N})$ leads to

$$\mathcal{E};\varnothing \vdash \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_\pi \sim \mathbf{if}\ \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \mathbf{then}\ \mathbf{c}'_{\mathrm{id}}$$

6. Now, let us focus on $\mathbf{c}'_\pi \overset{\text{def}}{=} \mathbf{shuffle}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ \pi\ (\mathbf{r}\ i_4)$. By the characterisation of the predicate **shuffle** given in the rule L.Shuffle, we have, for all $i \in [\![1; N]\!]$,

$$\mathbf{c}'_\pi \circledast \mathbf{i} = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast (\pi^{-1} \cdot \mathbf{i}))\ \langle \mathbf{r}\ i_4 \mid \pi^{-1} \cdot \mathbf{i} \rangle$$

By the function application rule G.∼:FA and structural rules for **if** b **then** t **else** u and n-tuples L.Comp, and because

$$\mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \rightarrow \bigwedge_{i=1}^{N} (\mathbf{valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \langle \mathbf{c} \mid \mathbf{i} \rangle\ v),$$

we have to prove that for all $i \in [\![1; N]\!]$, we have

$$\begin{aligned}\mathcal{E}; \varnothing \vdash \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v, \mathbf{if}\ \mathbf{valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \langle \mathbf{c} \mid (\pi^{-1} \cdot \mathbf{i})\rangle\ v \\ \mathbf{then}\ \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast (\pi^{-1} \cdot \mathbf{i}))\ \langle \mathbf{r}\ i_4 \mid (\pi^{-1} \cdot \mathbf{i})\rangle \\ \sim \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v, \mathbf{if}\ \mathbf{valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \langle \mathbf{c} \mid \mathbf{i}\rangle\ v \\ \mathbf{then}\ \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast \mathbf{i})\ \langle \mathbf{r}\ i_4 \mid \mathbf{i}\rangle\end{aligned}$$

Now, let $i \in [\![1; N]\!]$. By the action of $\circledast$ on canonical vectors rule L.$\circledast$:CanoVec, we have $\langle \mathbf{c} \mid \mathbf{i}\rangle = \mathbf{c} \circledast \mathbf{i}$. Therefore, by the rewrite rule L.Rewrite and by the indistinguishability rule for *shuffle-friendly* maps G.Sfm:IndCPA in both previous sides, we have to prove

$$\begin{aligned}\mathcal{E}; \Theta \vdash \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v, \mathbf{if}\ \mathbf{valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \langle \mathbf{c} \mid (\pi^{-1} \cdot \mathbf{i})\rangle\ v \\ \mathbf{then}\ \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{0}\ (\mathbf{len}\ (\mathbf{c} \circledast (\pi^{-1} \cdot \mathbf{i}))))\ \langle \mathbf{r}\ i_4 \mid (\pi^{-1} \cdot \mathbf{i})\rangle \\ \sim \mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v, \mathbf{if}\ \mathbf{valid}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \langle \mathbf{c} \mid \mathbf{i}\rangle\ v \\ \mathbf{then}\ \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{0}\ (\mathbf{len}\ (\mathbf{c} \circledast \mathbf{i})))\ \langle \mathbf{r}\ i_4 \mid \mathbf{i}\rangle\end{aligned}$$

However, by definition of the predicate $\mathbf{valid}_N$, we have the following formula

$$\mathbf{valid}_N\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \mathbf{c}\ v\ \rightarrow \bigwedge_{1 \leqslant i < j \leqslant N} (\mathbf{len}\ \langle \mathbf{c} \mid \mathbf{i}\rangle = \mathbf{len}\ \langle \mathbf{c} \mid \mathbf{j}\rangle).$$

Therefore, the property we have to show holds by reflexivity G.$\sim$:Refl.

D.2 Rewinding lemma proof

To prove the rewinding lemma, we will need the Chernoff bound, which we recall here:

Lemma D.1 (Chernoff bound)**.** Let $X_1, \ldots, X_n : \mathbb{N} \longrightarrow \{0, 1\}$ be n independent and identically distributed random variables, *i.e.* there exists a number $p \in [0, 1]$ such that, for all $i \in [\![1; n]\!]$, $\Pr[X_i = 1] = p$. Then we have

$$\forall\, \delta \in]0, 1[, \Pr\left[\ \sum_{i=1}^{n} X_i \leqslant (1 - \delta)np\ \right] \leqslant \exp\left(-\frac{\delta^2}{2}np\right).$$

Proposition D.1 (Rewinding lemma)**.** Let $\phi \stackrel{\text{def}}{=} \lambda x.\phi(x) : \tau \to \mathbf{bool}\ [\mathsf{ptime}]$ be a polynomial time property. Let $n \in \mathbb{N}^*$ be a natural number. Let g : **real** with **non-negl**(g) be a non-negligible parameter. Then we conclude the following judgement.

$$\begin{aligned}\mathcal{E}; \Theta \vdash \tilde{\exists}\, k : \mathbf{nat}.\ \mathbf{det}(k)\ \tilde{\wedge}\ \mathbf{pbound}(k) \mathrel{\tilde{\to}} \tilde{\exists}\, \mathbf{r}_s : \mathbf{nat} \to \tau. \tilde{\exists}\, \mathbf{select}^{(n)}_{\text{rand}} : \mathbf{nat} \to (\mathbf{nat} \to \tau) \to \mathbf{set}_n(\tau). \\ [\mathbf{low\text{-}bound}\ g\ \phi \to \forall\, (\mathbf{r}_s\ t : \tau) \in \mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s.\ \phi\ (\mathbf{r}_s\ t)]\tilde{\wedge}\ [\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]\end{aligned}$$

Proof. Let g : **real** with **non-negl**(g) be a non-negligible parameter. Let $\eta \in \mathbb{N}^*$ be a security parameter and let $\rho \in \Omega$ be a random tape. Let $Y_{\eta,\rho}$ be the following random variable

$$Y_{\eta,\rho}(i) \stackrel{\text{def}}{=} \phi_{\eta,\rho}(\mathbf{r}_s(i)) \in \{0, 1\}$$

Let $Y_1(\eta, \rho), \ldots, Y_n(\eta, \rho)$ be n consecutive draw of the random variables $Y_{\eta,\rho}$ for n index $i_j \in [\![1; k]\!]$ randomly chosen. As $\mathbf{r}_s$ is a source of *uniformly distributed and independent* random variables, the random variables $Y_j(\eta, \rho)$, for all $j \in [\![1; n]\!]$, are *mutually independent*. We suppose we have

$$p_Y(\eta, \rho) \stackrel{\text{def}}{=} \Pr_{r \in X}\left[\ \phi_{\eta,\rho}(r) = 1\ \right] \geqslant \mathbb{E}_{\rho' \in \Omega}\big([\![g]\!]^{\eta;\rho}_{\mathbb{M}:\mathcal{E}}\big).$$

We want to prove that the function $\eta \longmapsto \Pr_{\rho\in\Omega}\left[\ \sum_{j=1}^{k} Y_j(\eta,\rho) \geqslant n\ \right]$ is overwhelming. In fact, we will show that, for all security parameter $\eta \in \mathbb{N}^*$, $\Pr_{\rho\in\Omega}\left[\ \sum_{j=1}^{k} Y_j(\eta,\rho) < n\ \right] \leqslant \frac{1}{2^\eta}$, which is equivalent to the property we want to show. To prove this property, we will use the Chernoff bound which states the following property

$$\forall\, \delta \in]0,1[, \Pr_{\rho\in\Omega}\left[\ \sum_{j=1}^{k} Y_j(\eta,\rho) \leqslant (1-\delta)kp_Y(\eta,\rho)\ \right] \leqslant \exp\left(-\frac{\delta^2}{2}kp_Y(\eta,\rho)\right).$$

Therefore, to obtain the property we want, we have to find a pair $(\delta(\eta), k(\eta)) \in]0,1[\times\mathbb{N}^*$ such that

$$(1-\delta(\eta))k(\eta)p_Y(\eta,\rho) < n \quad \text{and} \quad \exp\left(-\frac{\delta(\eta)^2}{2}k(\eta)p_Y(\eta,\rho)\right) \leqslant \frac{1}{2^\eta} \tag{$\mathcal{I}$}$$

By monotonic increasing of the logarithm function, the second equation becomes

$$\frac{\delta(\eta)^2}{2}k(\eta)p_Y(\eta,\rho) \geqslant \eta\ln 2.$$

In fact, the system of inequalities Eq. ($\mathcal{I}$) can be solved by solving the following system of equations where we have to find a pair $(\delta(\eta), x(\eta)) \in]0,1[\times\mathbb{R}^+$ such that

$$(1-\delta(\eta))x(\eta)p_Y(\eta,\rho) = n \tag{1}$$

$$\text{and} \quad \frac{\delta(\eta)^2}{2}x(\eta)p_Y(\eta,\rho) = \eta\ln 2. \tag{2}$$

Indeed, if we have found a solution $(\delta(\eta), x(\eta))$ of the second system of equations, the pair $(\delta(\eta), \lceil x(\eta)\rceil)$ is a solution of the first system Eq. ($\mathcal{I}$). The second equation Eq. (2) leads to

$$x(\eta) = \frac{2\eta\ln 2}{\delta(\eta)^2 p_Y(\eta,\rho)}. \tag{$*$}$$

Hence, by equations Eq. ($*$) and Eq. (1) leads to the following quadratic equation

$$n\delta(\eta)^2 + (2\eta\ln 2)\delta(\eta) - 2\eta\ln 2 = 0. \tag{E_δ}$$

The solutions of this quadratic equation are given by

$$\delta_\pm(\eta) \stackrel{\text{def}}{=} \frac{-2\eta\ln 2 \pm \sqrt{\Delta}}{2n}$$

where $\Delta = (2\eta\ln 2)^2 + 8n\eta\ln 2 > 0$. Moreover, we have $\delta_-(\eta) < 0$ and $\delta_+(\eta) > 0$. Besides, we have

$$\begin{aligned}
\delta_+(\eta) < 1 \quad &\Longleftrightarrow \quad \sqrt{1+\frac{2n}{\eta\ln 2}} < \frac{n}{\eta\ln 2} + 1 \\
&\Longleftrightarrow \quad 1 + \frac{2n}{\eta\ln 2} < \left(\frac{n}{\eta\ln 2} + 1\right)^2 \\
&\Longleftrightarrow \quad \left(\frac{n}{\eta\ln 2}\right)^2 > 0.
\end{aligned}$$

Therefore, only the solution $\delta_+(\eta)$ interest us and the partnered solution $x(\eta)$ is given by

$$x(\eta) = \frac{2n^2}{\eta p_Y(\eta,\rho)\ln 2}\left(1 - \sqrt{1+\frac{2n}{\eta\ln 2}}\right)^{-2}.$$

Therefore, we denote by $f_n : \mathbb{N}^* \longrightarrow \mathbb{R}_+^*$ such that

$$\forall\, \eta \in \mathbb{N}^*, x(\eta) \stackrel{\text{def}}{=} \frac{f_n(\eta)}{p_Y(\eta, \rho)}.$$

To conclude, we have to study the asymptotic behavior of the function f_n, to show this function is at least polynomial bounded in the security parameter η. By series expansion, we have the following results.

$$\begin{aligned}
\forall\, \eta \in \mathbb{N}^*, f_n(\eta) &= \frac{n^2}{\eta \ln 2}\left(1 - \sqrt{1 + \frac{2n}{\eta \ln 2}} + \frac{n}{\eta \ln 2}\right)^{-1} \\
&= \frac{n^2}{\eta \ln 2}\left(2\left(\frac{n}{2 \ln 2}\right)^2 \frac{1}{\eta^2} + o_{\eta \to +\infty}\left(\tfrac{1}{\eta^2}\right)\right)^{-1} \\
&= 2(\ln 2)\eta\left(1 + o_{\eta \to +\infty}(1)\right).
\end{aligned}$$

Therefore, the asymptotic analysis of function f_n gives us the following result

$$\boxed{f_n(\eta) \sim_{\eta \to +\infty} 2(\ln 2)\eta}. \tag{Θ}$$

Moreover, by hypothesis on $p_Y(\eta, \rho)$ given by the hypothesis
low-bound $g\ \phi$, we have $x(\eta) \leqslant \frac{f_n(\eta)}{\mathbb{E}_{\rho \in \Omega}\left([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\right)}$. Therefore, if we denote by $k(\eta) \in \mathbb{N}^*$ the quantity

$$\boxed{k(\eta) = \left\lceil \frac{f_n(\eta)}{\mathbb{E}_{\rho}\left([\![g]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\right)} \right\rceil},$$

we conclude, as g is a non-negligible parameter and because of result Eq. (Θ) that k is polynomial in the security parameter η.

Proof assessment We have proved that if k : **nat** is the natural term for whose semantics is given by

$$\forall\, \eta \in \mathbb{N}^*, \forall\, \rho \in \Omega, [\![k]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \stackrel{\text{def}}{=} \left\lceil \frac{1}{\mathbb{E}_{\rho'}\left([\![g]\!]^{\eta,\rho'}_{\mathbb{M}:\mathcal{E}}\right)} \frac{2n^2}{\eta \ln 2}\left(1 - \sqrt{1 + \frac{2n}{\eta \ln 2}}\right)^{-2} \right\rceil$$

then k is polynomial in the security parameter η and is deterministic. Moreover, by analysis of the system Eq. ($\mathcal{I}$), if we denote by Φ and $\mathcal{H}$ the functions respectively defined by

$$\begin{aligned}
\Phi(\eta, \rho) &\stackrel{\text{def}}{=} [\![\forall\, (\mathbf{r}_s\ t : \tau) \in \mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s.\ \phi\ (\mathbf{r}_s\ t)]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}} \\
\text{and} \quad \mathcal{H}(\eta, \rho) &\stackrel{\text{def}}{=} [\![\mathbf{low\text{-}bound}\ g\ \phi]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}},
\end{aligned}$$

we have shown the following result

$$\forall\, \eta \in \mathbb{N}^*, \Pr_{\rho \in \Omega}\left[\ \Phi(\eta, \rho)\ \middle|\ \mathcal{H}(\eta, \rho)\ \right] \geqslant 1 - \frac{1}{2^\eta}.$$

And finally, by definition of the function $\mathbf{select}^{(n)}_{\text{rand}}$ given in Algorithm 2, for all $\mathbf{r}_s : \mathbb{N}^* \longrightarrow X$, we have $\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s(i)\}_{i=1}^k$ and then we conclude

$$\forall\, \eta \in \mathbb{N}^*, \Pr_{\rho \in \Omega}\left[\ [\![\mathbf{select}^{(n)}_{\text{rand}}\ k\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k\}]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}\ \right] = 1.$$

Consequently, those results achieves the proof of the rewinding lemma. □

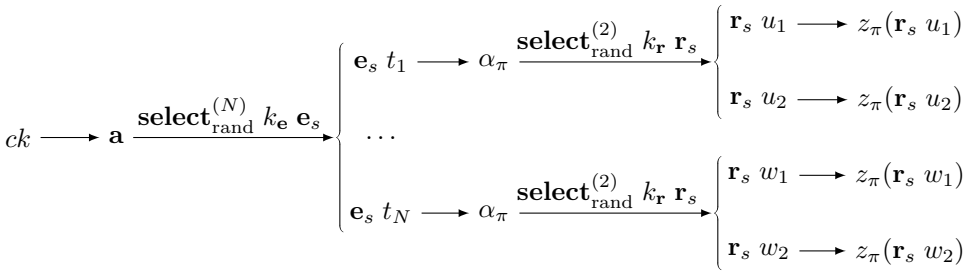

Figure 9: Skeleton of commited matrix extraction proof

D.3 Verifiability proof

Let $\phi \stackrel{\text{def}}{=} ck\ n, \mathbf{a}, \mathbf{e}_\pi, \alpha_\pi, r_\pi, z_\pi(r_\pi), sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi, \alpha_\phi, r_\phi, z_\phi(r_\phi)$ be a trace such that

$$\textbf{zkp-verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_\pi)\ \langle \alpha_\pi, r_\pi, z_\pi(r_\pi)\rangle \wedge\ \textbf{zkp-verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r_\phi, z_\phi(r_\phi)\rangle \wedge\ \textbf{wf\_ctxt}_N\ sk\ \mathbf{c}.$$

D.3.1 Extraction of the commited matrix

To be able to rebuild the commited matrix, we have to extract N witnesses $(\mathbf{e}'_i, k_i)_{i=1}^N$ for the relations of correct commitment $\mathcal{R}^{\text{com}}(\mathbf{e}_i)$, where $(\mathbf{e}_i)_{i=1}^N$ is a free family of $\mathbb{Z}_{q_\eta}^N$. Consequently, there is two steps of rewinding, one on the vectors $\mathbf{e}_i$, for $i \in [\![1; N]\!]$ and the other one is when we obtain a candidate vector $\mathbf{e}_i$, we have to rewind the challenge $r \in \mathbb{Z}_{q_\eta}$ to be able to use the *special-soundness* axiom. Therefore, in that case, we have to use two times the predicate **low-bound**, one states there is enough random vectors to rewind and the second one states that for a chosen vector, there is enough random challengs to rewind. Hence, if we denote by ψ_π the formula

$$\psi_\pi \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \lambda r.\ \textbf{zkp-verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r)\rangle,$$

we have to suppose the following property

$$\textbf{low-bound}\ g\ (\lambda \mathbf{e}.\ \textbf{low-bound}\ g'\ (\psi_\pi\ \mathbf{e}))$$

for two parameters $g, g' : \textbf{real}$ with $\textbf{non-negl}(g)$ and $\textbf{non-negl}(g')$.

Lemma D.2. Let $\mathcal{E}$ be an environment, let Θ be a context of global formulas and let Γ be a context of local formulas. We denote by ψ_π the formula

$$\psi_\pi \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \lambda r.\textbf{zkp-verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r)\rangle.$$

We suppose

$$\mathcal{E}; \Theta; \Gamma \vdash \textbf{low-bound}\ g\ (\lambda \mathbf{e}.\ \textbf{low-bound}\ g'\ (\psi_\pi\ \mathbf{e})), \qquad (\mathcal{H}_{\mathbf{e},r})$$

with

$$\mathcal{E}; \Theta \vdash \textbf{non-negl}(g)\ \tilde{\wedge}\ \textbf{det}(g) \quad \text{and} \quad \mathcal{E}; \Theta \vdash \textbf{non-negl}(g')\ \tilde{\wedge}\ \textbf{det}(g')$$

Then, the property $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} = \textbf{com-mat}\ (ck\ n)\ M\ \mathbf{s}$ holds, where there exists a name $\mathbf{e}_s : \textbf{nat} \to \textbf{msg}$ and N terms $t_1, \ldots, t_N : \textbf{nat}$ pairwise distincts such that there exists a name $\mathbf{r}_s : \textbf{nat} \to \textbf{msg}$ and 2 terms $r_{i,1}, r_{i,2} : \textbf{nat}$ with $r_{i,1} \neq r_{i,2}$ such that if we denote, for all $i \in [\![1; N]\!]$, $\mathbf{e}'_i \stackrel{\text{def}}{=} \pi_2\ w_\pi(i)$ and $k_i \stackrel{\text{def}}{=} \pi_3\ w_\pi(i)$ with

$$w_\pi(i) \stackrel{\text{def}}{=} \textbf{zkp-extract}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_s\ t_i)\ \langle \alpha_\pi, \mathbf{r}_s\ r_{i,1}, z_\pi(\mathbf{r}_s\ r_{i,1})\rangle\ \langle \alpha_\pi, \mathbf{r}_s\ r_{i,2}, z_\pi(\mathbf{r}_s\ r_{i,2})\rangle$$

then terms M and $\mathbf{s}$ are defined by $M \stackrel{\text{def}}{=} \pi_1\ u$ and $\mathbf{s} \stackrel{\text{def}}{=} \pi_2\ u$ where $u \stackrel{\text{def}}{=} \textbf{solve}\ \mathbf{a}\ (\mathbf{e}_s\ t_i)_{i=1}^N\ (\mathbf{e}'_i, k_i)_{i=1}^N$.

Proof. Firstly, we have to obtain N vectors such that the adversary produces at least two different proof transcripts but for the same commit message to be able to apply the *special-soundness* axiom. To do so, we apply the rewinding lemma (Proposition 5.2) to the formula $\psi \overset{\text{def}}{=} \lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e})$. Hence, there exists a polynomial bounded and deterministic term $k_{\mathbf{e}} : \mathbf{nat}$ such that $N \leqslant k_{\mathbf{e}}$, a name $\mathbf{e}_s : \mathbf{nat} \to \mathbf{msg}$, and a selection function $\mathbf{select}^{(N)}_{\text{vect}}$ such that

$$\mathcal{E}; \Theta \vdash [\mathbf{low\text{-}bound}\ g\ \psi \to \forall\, (\mathbf{e}_s\ t) \in \mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s.\ \psi\ (\mathbf{e}_s\ t)] \tilde{\wedge}\ [\mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s \subseteq \{\mathbf{e}_s\ 1, \ldots, \mathbf{e}_s\ k_{\mathbf{e}}\}]$$

Therefore, by L.ByGlob and by hypothesis Eq. ($\mathcal{H}_{\mathbf{e},r}$), we conclude

$$\mathcal{E}; \Theta; \Gamma \vdash \forall\, (\mathbf{e}_s\ t) \in \mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s.\ \psi\ (\mathbf{e}_s\ t).$$

On another hand, we have the following global formula by the rule L.Basis

$$\mathcal{E}; \Theta \vdash [\mathbf{basis}_N\ (\mathbf{e}_s\ i)_{i=1}^N].$$

Therefore, by the second conclusion of the rewinding lemma and by the transfer of properties by adversarial selection rule G.Sel, we have

$$\mathcal{E}; \Theta \vdash [N \leqslant k_{\mathbf{e}} \to \mathbf{basis}_N\ (\mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s)] \tag{β}$$

Moreover, by the second conclusion of the rewinding lemma, and because $\text{Card}([\![\mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}) = N$ by definition of the semantics of the type $\mathbf{set}_N(\mathbf{msg})$, we conclude the existence of N pairwise distinct terms $t_1, \ldots, t_N : \mathbf{nat}$ such that $1 \leqslant t_1 < \ldots < t_N \leqslant k_{\mathbf{e}}$ (without loss of generality for the order of terms t_i) and $\mathbf{select}^{(N)}_{\text{vect}}\ k_{\mathbf{e}}\ \mathbf{e}_s = \{\mathbf{e}_s\ t_i\}_{i=1}^N$. Therefore, for all $i \in [\![1; N]\!]$, we have

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ (\mathbf{e}_s\ t_i)) \tag{$\mathcal{H}_r$}$$

Now we have obtain those N vectors, we can apply the rewinding lemma for each vector to obtain two different proof transcripts but for the same commit message in the goal of extract a witness by the *special-soundness* property. Let $i \in [\![1; N]\!]$. By the rewinding lemma (Proposition 5.2) applied to the formula $\psi_\pi\ (\mathbf{e}_s\ t_i)$ the existency of a polynomial bounded and deterministic term $k_r : \mathbf{nat}$ such that $2 \leqslant k_r$, a name $\mathbf{r}_s : \mathbf{nat} \to \mathbf{msg}$, and a selection function $\mathbf{select}^{(2)}_{\text{chall}}$ such that

$$\mathcal{E}; \Theta \vdash [\mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ (\mathbf{e}_s\ t_i)) \to \forall\, (\mathbf{r}_s\ t) \in \mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s.\ \psi_\pi\ (\mathbf{e}_s\ t_i)\ (\mathbf{r}_s\ t)] \tilde{\wedge}\ [\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k_r\}]$$

Therefore, by L.ByGlob and by hypothesis Eq. ($\mathcal{H}_r$), we conclude

$$\mathcal{E}; \Theta; \Gamma \vdash \forall\, (\mathbf{r}_s\ t) \in \mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s.\ \psi_\pi\ (\mathbf{e}_s\ t_i)\ (\mathbf{r}_s\ t).$$

By the second conclusion of the rewinding lemma, and because $\text{Card}([\![\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}}) = 2$, we conclude the existency of 2 distinct terms $r_{i,1}, r_{i,2} : \mathbf{nat}$ with, without loss of generality, $1 \leqslant r_{i,1} < r_{i,2} \leqslant k_r$ and $\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s = \{\mathbf{r}_s\ r_{i,1}, \mathbf{r}_s\ r_{i,2}\}$. Therefore, for all $i \in [\![1; N]\!]$ and for all $j \in \{1, 2\}$, we have

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_s\ t_i)\ \langle \alpha_\pi, \mathbf{r}_s\ r_{i,j}, z_\pi(\mathbf{r}_s\ r_{i,j})\rangle$$

By the *special-soundness* property L.Σ-P:SpSound applied to the relation of correct commitment $\mathcal{R}^{\text{com}}([\![\mathbf{e}_s\ t_i]\!]^{\eta,\rho}_{\mathbb{M}:\mathcal{E}})$, we conclude the existency of an extractor function $\mathbf{zkp\text{-}extract}_\pi$ such that

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{zkp\text{-}rel}_\pi\ x_\pi(i)\ (\mathbf{zkp\text{-}extract}_\pi\ x_\pi(i)\ \mathbf{p}_\pi(i,1)\ \mathbf{p}_\pi(i,2))$$

where $x_\pi(i) \overset{\text{def}}{=} (ck\ n, \mathbf{a}, \mathbf{e}_s\ t_i)$ and $\mathbf{p}_\pi(i,j) \overset{\text{def}}{=} \langle \alpha_\pi, \mathbf{r}_s\ r_{i,j}, z_\pi(\mathbf{r}_s\ r_{i,j})\rangle$, for all $i \in [\![1; N]\!]$ and $j \in \{1, 2\}$. Hence, for all $i \in [\![1; N]\!]$, we denote by $w_\pi(i)$ the witness $w_\pi(i) \overset{\text{def}}{=} \mathbf{zkp\text{-}extract}_\pi\ x_\pi(i)\ \mathbf{p}_\pi(i,1)\ \mathbf{p}_\pi(i,2)$. Besides, let

$\mathbf{e}'_i$ and k_i be the terms defined by $\mathbf{e}'_i \overset{\text{def}}{=} \pi_2\ w_\pi(i)$ and $k_i \overset{\text{def}}{=} \pi_3\ w_\pi(i)$. Therefore, by definition of the predicate $\mathbf{zkp\text{-}rel}_\pi$, we have in particular

$$\mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{i=1}^{N} (\mathbf{a} \circledast (\mathbf{e}_s\ t_i) = \mathbf{com\text{-}vec}\ (ck\ n)\ \mathbf{e}'_i\ k_i). \tag{$*$}$$

Hence, by properties Eq. (β), Eq. ($*$) and by the commitment opening rule L.Open, we conclude

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s}$$

where $M \overset{\text{def}}{=} \pi_1\ v$ and $\mathbf{s} \overset{\text{def}}{=} \pi_2\ v$ with $v \overset{\text{def}}{=} \mathbf{solve}\ \mathbf{a}\ (\mathbf{e}_s\ t_i)_{i=1}^N\ (\mathbf{e}'_i, k_i)_{i=1}^N$. □

D.3.2 M represents a permutation

Lemma D.3. Let $\mathcal{E}$ be an environment, let Θ be a context of global formulas and let Γ be a context of local formulas. We denote by ψ the function $\psi \overset{\text{def}}{=} \lambda r.\ \mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_\pi)\ \langle \alpha_\pi, r, z_\pi(r) \rangle$. We suppose

$$\mathcal{E}; \Theta \vdash \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g) \tag{$\mathcal{H}_g$}$$
$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{low\text{-}bound}\ g\ \psi, \tag{$\mathcal{H}_1$}$$
$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s}, \tag{$\mathcal{H}_2$}$$

Then, we conclude $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{perm}_N\ M$.

Proof. By the rewinding lemma (Proposition 5.2) applied to the formula ψ, their exists a polynomial bounded and deterministic term $k_r : \mathbf{nat}$ such that $2 \leqslant k_r$, a name $\mathbf{r}_s : \mathbf{nat} \to \mathbf{msg}$, and a selection function $\mathbf{select}^{(2)}_{\text{chall}}$ such that

$$\mathcal{E}; \Theta \vdash [\mathbf{low\text{-}bound}\ g\ \psi \to \forall\, (\mathbf{r}_s\ t) \in \mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s.\ \psi\ (\mathbf{r}_s\ t)] \tilde{\wedge}\ [\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s \subseteq \{\mathbf{r}_s\ 1, \ldots, \mathbf{r}_s\ k_r\}]$$

Therefore, by L.ByGlob and by hypothesis Eq. ($\mathcal{H}_1$), we conclude

$$\mathcal{E}; \Theta; \Gamma \vdash \forall\, (\mathbf{r}_s\ t) \in \mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s.\ \psi\ (\mathbf{r}_s\ t).$$

Hence, their exists 2 distinct terms $r_1, r_2 : \mathbf{nat}$ with, without loss of generality, $1 \leqslant r_1 < r_2 \leqslant k_r$ and $\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s = \{r_1, r_2\}$. Therefore, we have

$$\mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{j \in \{1,2\}} (\mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_\pi)\ \langle \alpha_\pi, \mathbf{r}_s\ r_j, z_\pi(\mathbf{r}_s\ r_j) \rangle).$$

By the *special-soundness* property L.Σ-P:SpSound applied to the relation of correct commitment $\mathcal{R}^{\text{com}}(\llbracket \mathbf{e}_\pi \rrbracket^{\eta,\rho}_{\mathbb{M}:\mathcal{E}})$, we conclude the existency of an extractor function $\mathbf{zkp\text{-}extract}_\pi$ such that

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{zkp\text{-}rel}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_\pi)\ w_\pi.$$

where w_π is the witness defined by

$$w_\pi \overset{\text{def}}{=} \mathbf{zkp\text{-}extract}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e}_\pi)\ \langle \alpha_\pi, \mathbf{r}_s\ r_1, z_\pi(\mathbf{r}_s\ r_1) \rangle\ \langle \alpha_\pi, \mathbf{r}_s\ r_2, z_\pi(\mathbf{r}_s\ r_2) \rangle.$$

Hence, let t, $\mathbf{e}'$ and k be the terms defined respectively by $t \overset{\text{def}}{=} \pi_1\ w_\pi$, $\mathbf{e}' \overset{\text{def}}{=} \pi_2\ w_\pi$ and $k \overset{\text{def}}{=} \pi_3\ w_\pi$. By definition of the correct commitment relation predicate $\mathbf{zkp\text{-}rel}_\pi$, we have the three following properties

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} \circledast \mathbf{1} = \mathbf{com\text{-}vec}\ (ck\ n)\ \mathbf{1}\ t \tag{i}$$
$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} \circledast \mathbf{e}_\pi = \mathbf{com\text{-}vec}\ (ck\ n)\ \mathbf{e}'\ k \tag{ii}$$
$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{prod}_N\ \mathbf{e}' = \mathbf{prod}_N\ \mathbf{e}_\pi \tag{iii}$$

– By Eq. (i), by hypothesis Eq. $(\mathcal{H}_2)$ and by using the rewrite rule L.Rewrite, we have $\mathcal{E};\Theta;\Gamma \vdash (\mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s}) \circledast \mathbf{1} = \mathbf{com\text{-}vec}\ (ck\ n)\ \mathbf{1}\ t$. Next, by action of $\circledast$ on commitments and by transitivity, the rule L.$\circledast$:Com applied to the previous identity leads to $\mathcal{E};\Theta;\Gamma \vdash \mathbf{com\text{-}vec}\ (ck\ n)\ (M \cdot \mathbf{1})\ \langle \mathbf{s} \mid \mathbf{1}\rangle = \mathbf{com\text{-}vec}\ (ck\ n)\ \mathbf{1}\ t$. Finally, as the commitment scheme $\mathbb{KS}(\mathbb{Z}_{q_\eta}^N)$ is *computationally binding*, we conclude, thanks to the related rule L.Com:Bind, the following equality

$$\mathcal{E};\Theta;\Gamma \vdash M \cdot \mathbf{1} = \mathbf{1}. \tag{$*_{\mathbf{1}}$}$$

– Similarly, using equation Eq. (ii) and hypothesis Eq. $(\mathcal{H}_2)$, we conclude by the *binding* rule L.Com:Bind the following judgement $\mathcal{E};\Theta;\Gamma \vdash M \cdot \mathbf{e}_\pi = \mathbf{e}'$. Hence, by the rewrite rule L.Rewrite and by the last equation Eq. (iii), the last identity leads to

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{prod}_N\ (M \cdot \mathbf{e}_\pi) = \mathbf{prod}_N\ \mathbf{e}_\pi.$$

Let $P_N(M)$ be the polynomial defined by $P_N(M) \stackrel{\text{def}}{=} \mathbf{prod}_N\ (M \cdot X) - \mathbf{prod}_N\ X$. As $\mathbf{e}_\pi$ is a fresh name, we can apply the *Schwartz-Zippel* lemma to the polynomial $P_N(M)$ and conclude by the related rule L.Sz

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{prod}_N\ (M \cdot X) = \mathbf{prod}_N\ X. \tag{$*_\Pi$}$$

Consequently, as equations Eq. $(*_{\mathbf{1}})$ and Eq. $(*_\Pi)$ hold, we conclude by the characterisation of permutation matrix that M represents a permutation, *i.e.* by apply the rule L.π:Charac, the following judgement holds $\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ M$. Therefore, the vector $\mathbf{a}$ is a commit message to a permutation matrix, *i.e.* we have

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s} \quad \text{and} \quad \mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ M.$$

□

D.3.3 M was used to shuffle the inputted ciphertexts list with the *shuffle-friendly* map ϕ_{pk}

Lemma D.4. Let $\mathcal{E}$ be an environment, let Θ be a context of global formulas and let Γ be a context of local formulas. We denote by ψ_ϕ the formula defined by

$$\psi_\phi \stackrel{\text{def}}{=} \lambda r.\ \mathbf{zkp\text{-}verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r, z_\phi(r)\rangle$$

We suppose

$$\mathcal{E};\Theta \vdash \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g) \tag{$\mathcal{H}_g$}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{low\text{-}bound}\ g\ \psi_\phi \tag{$\mathcal{H}_r$}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ M\ \mathbf{s} \tag{$\mathcal{H}_{\mathbf{a}}$}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ M \tag{$\mathcal{H}_\pi$}$$

Then, we conclude the following property

$$\mathcal{E};\Theta;\Gamma \vdash \bigwedge_{i=1}^{N} (\exists\, v_i.\ \mathbf{c}' \circledast (M \cdot \mathbf{i}) = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast \mathbf{i})\ v_i).$$

Proof. To ease notations, we denote by x_ϕ the statement defined by $x_\phi \stackrel{\text{def}}{=} (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)$. By hypothesis Eq. $(\mathcal{H}_r)$ and by the rewinding lemma (Proposition 5.2) applied to the function ψ_ϕ, we conclude the existency of a term $k_r : \mathbf{nat}$ such that $2 \leqslant k_r$, a name $\mathbf{r}_s : \mathbf{nat} \to \mathbf{msg}$ and a selection function $\mathbf{select}^{(2)}_{\text{chall}}$ such that their exists 2 distinct terms $t_1, t_2 : \mathbf{nat}$ with $1 \leqslant t_1 < t_2 \leqslant k_r$ such that $\mathbf{select}^{(2)}_{\text{chall}}\ k_r\ \mathbf{r}_s = \{\mathbf{r}_s\ t_1, \mathbf{r}_s\ t_2\}$ and

$$\mathcal{E};\Theta;\Gamma \vdash \bigwedge_{j\in\{1,2\}} (\mathbf{zkp\text{-}verif}_\phi\ x_\phi\ \langle \alpha_\phi, \mathbf{r}_s\ t_j, z_\phi(\mathbf{r}_s\ t_j)\rangle).$$

By the *special-soundness* property L.Σ-P:SpSound applied to the relation of correct shuffle $\mathcal{R}_{\phi_{pk}}^{\text{shuffle}}([\![\mathbf{e}_\phi]\!]_{\mathbb{M}:\mathcal{E}}^{\eta;\rho})$, we conclude the existency of an extractor function $\mathbf{zkp\text{-}extract}_\phi$ such that

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{zkp\text{-}rel}_\phi\ x_\phi\ w_\phi$$

where w_ϕ is the witness defined by

$$w_\phi \stackrel{\text{def}}{=} \mathbf{zkp\text{-}extract}_\phi\ x_\phi\ \langle \alpha_\phi, \mathbf{r}_s\ t_1, z_\phi(\mathbf{r}_s\ t_1)\rangle\ \langle \alpha_\phi, \mathbf{r}_s\ t_2, z_\phi(\mathbf{r}_s\ t_2)\rangle.$$

Hence, let $\mathbf{e}'$, k and u be the terms defined respectively by $\mathbf{e}' \stackrel{\text{def}}{=} \pi_1\ w_\phi$, $k \stackrel{\text{def}}{=} \pi_2\ w_\phi$ and $u \stackrel{\text{def}}{=} \pi_3\ w_\phi$. By definition of the correct shuffle relation predicate $\mathbf{zkp\text{-}rel}_\phi$, we have the two following properties

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{a} \circledast \mathbf{e}_\phi = \mathbf{com\text{-}vec}\ (ck\ n)\ \mathbf{e}'\ k \tag{i}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{c}' \circledast \mathbf{e}' = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast \mathbf{e}_\phi)\ u \tag{ii}$$

By the first equation Eq. (i), by the hypothesis Eq. ($\mathcal{H}_\mathbf{a}$) and by the *binding* rule L.Com:Bind applied to the commitment scheme $\mathbb{KS}(\mathbb{Z}_{q_\eta}^N)$, we conclude $\mathcal{E};\Theta;\Gamma \vdash M \cdot \mathbf{e}_\phi = \mathbf{e}'$. Therefore, the second equation Eq. (ii) becomes

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{c}' \circledast (M \cdot \mathbf{e}_\phi) = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast \mathbf{e}_\phi)\ u$$

Besides, as $\mathbf{e}_\phi$ is a fresh name, we have $\mathcal{E};\Theta;\Gamma \vdash \phi_{\text{rand}}^{\mathbf{e}_\phi}(\mathbf{c}, \mathbf{c}', M)$. Moreover, by hypothesis Eq. ($\mathcal{H}_\pi$), M is a permutation matrix, *i.e.* the following property holds $\mathcal{E};\Theta;\Gamma \vdash \mathbf{perm}_N\ M$. Thus, by characterisation of *shuffle-friendly* maps given by the rule L.Sfm:Charac, the following property holds

$$\mathcal{E}, (\mathbf{x} : \mathbf{msg});\Theta;\Gamma \vdash \exists\, v_\mathbf{x}.\ \mathbf{c}' \circledast (M \cdot \mathbf{x}) = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast \mathbf{x})\ v_\mathbf{x}$$

In particular, this property holds for all vectors $\mathbf{i}$ where $i \in [\![1; N]\!]$ and achieve this way the proof. □

D.3.4 Proof of the verifiability property under conditions

Now we have obtain the 3 key lemmas to show that we can extract a permutation matrix π from the commit message $\mathbf{a}$ sent by the adversary and show that this matrix π was indeed used to *shuffle* the inputted ciphertexts list $\mathbf{c}$ to form the outputted ciphertexts list $\mathbf{c}'$, we can present the lemma proving the verifiability property we want but *under* some conditions needed to rewind parts of the protocol trace.

Lemma D.5. Let $\mathcal{E}$ be an environment, let Θ be a context of global formulas and let Γ be a context of local formulas. We denote by $\mathcal{H}$ the function defined by

$$\mathcal{H} \stackrel{\text{def}}{=} \lambda\mathbf{e}.\ \lambda r.\ \lambda r'.\ \mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r)\rangle \wedge \mathbf{zkp\text{-}verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r', z_\phi(r')\rangle \wedge \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}.$$

We suppose

$$\mathcal{E};\Theta \vdash \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g) \tag{$\mathcal{H}_g$}$$
$$\mathcal{E};\Theta \vdash \mathbf{non\text{-}negl}(g')\ \tilde{\wedge}\ \mathbf{det}(g') \tag{$\mathcal{H}_{g'}$}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{low\text{-}bound}\ g\ (\lambda\mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e})) \tag{$\mathcal{H}_1$}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e}_\pi) \tag{$\mathcal{H}_2$}$$
$$\mathcal{E};\Theta;\Gamma \vdash \mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \tag{$\mathcal{H}_3$}$$

Therefore, we conclude the following property

$$\mathcal{E};\Theta;\Gamma \vdash \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}' \wedge \mathbf{eqm}_N\ (\mathbf{dec\text{-}list}_{\mathbb{CS}}^{(N)}\ sk\ \mathbf{c})\ (\mathbf{dec\text{-}list}_{\mathbb{CS}}^{(N)}\ sk\ \mathbf{c}').$$

Proof. Let ψ_π be the formula defined by
$\psi_\pi \overset{\text{def}}{=} \lambda \mathbf{e}.\ \lambda r.\ \mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r)\rangle$. By definition of ψ_π and $\mathcal{H}$, we have the following global judgement

$$\mathcal{E}; \Theta \vdash {}_1[\mathcal{H}\ \mathbf{e}\ r\ r' \to \psi_\pi\ \mathbf{e}\ r]. \tag{$*_\pi$}$$

Hence, by the rule L.LB:Trans of local transitivity and because $\mathcal{E} \vdash g' : \mathbf{real}$ with $\mathcal{E}; \Theta \vdash \mathbf{non\text{-}negl}(g)$, we conclude $\mathcal{E}; \Theta; \varnothing \vdash \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e}) \to \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e})$ for all vector $\mathbf{e}$. In fact, this last property is true with probability 1, *i.e.* we have $\mathcal{E}; \Theta \vdash {}_1[\mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e}) \to \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e})]$. Therefore, by apply one more time the rule L.LB:Trans and because $\mathcal{E} \vdash g : \mathbf{real}$ and $\mathcal{E}; \Theta \vdash \mathbf{non\text{-}negl}(g)$, we conclude

$$\mathcal{E}; \Theta; \varnothing \vdash \mathbf{low\text{-}bound}\ g\ (\lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e})) \to \mathbf{low\text{-}bound}\ g\ (\lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e})).$$

Hence, by this last judgement and by hypothesis Eq. ($\mathcal{H}_1$), we conclude

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{low\text{-}bound}\ g\ (\lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e})).$$

Therefore, by the first key lemma (Lemma D.2), we conclude the existency of two terms π and $\mathbf{s}$ such that $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ \pi\ \mathbf{s}$.

Next, by global property Eq. ($*_\pi$), by the local transitivity rule L.LB:Trans, because $\mathcal{E} \vdash g' : \mathbf{real}$ and $\mathcal{E}; \Theta \vdash \mathbf{non\text{-}negl}(g')$ and by hypothesis Eq. ($\mathcal{H}_2$), we conclude $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{low\text{-}bound}\ g'\ (\psi_\pi\ \mathbf{e}_\pi)$. Therefore, by the second key lemma (Lemma D.3), the rebuild matrix π previously obtained is a permutation matrix, *i.e.* we have the property $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{perm}_N\ \pi$. Hence the vector $\mathbf{a}$ sends by the adversary can be open to a permutation matrix:

$$\mathcal{E}; \Theta; \Gamma \vdash \mathbf{a} = \mathbf{com\text{-}mat}\ (ck\ n)\ \pi\ \mathbf{s} \quad \text{and} \quad \mathcal{E}; \Theta; \Gamma \vdash \mathbf{perm}_N\ \pi. \tag{Γ}$$

Now, let ψ_ϕ be the formula defined by

$$\psi_\phi \overset{\text{def}}{=} \lambda r'.\ \mathbf{zkp\text{-}verif}_\phi\ x_\phi\ \langle \alpha_\phi, r', z_\phi(r')\rangle$$

where x_ϕ is the statement defined by $x_\phi \overset{\text{def}}{=} (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)$. Hence, by definition of ψ_ϕ and $\mathcal{H}$, we have the following global judgement

$$\mathcal{E}; \Theta \vdash {}_1[\mathcal{H}\ \mathbf{e}\ r\ r' \to \psi_\phi\ r']. \tag{$*_\phi$}$$

Hence, by global property Eq. ($*_\phi$), by the local transtivity rule L.LB:Trans, because $\mathcal{E} \vdash g' : \mathbf{real}$ and $\mathcal{E}; \Theta \vdash \mathbf{non\text{-}negl}(g')$ and by the second hypothesis Eq. ($\mathcal{H}_2$), we conclude $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{low\text{-}bound}\ g'\ \psi_\phi$. Therefore, by this last property and by the conclusion Eq. (Γ), we can apply the third key lemma (Lemma D.4) and conclude the following property

$$\mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{i=1}^{N} (\exists\, v_i.\ \mathbf{c}' \circledast (\pi \cdot \mathbf{i}) = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ (\mathbf{c} \circledast \mathbf{i})\ v_i).$$

By conclusion $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{perm}_N\ \pi$ given by Eq. (Γ) and by the injectivity rule for permutations L.π:Inj, we have, for all $i \in [\![1; N]\!]$, the existency of an index $j_i \in [\![1; N]\!]$ such that the property $\mathcal{E}; \Theta; \Gamma \vdash \pi \cdot \mathbf{i} = \mathbf{j_i}$ holds. Hence, by the action rule of $\circledast$ on canonical vectors applied to $\mathbf{i}$ and $\mathbf{j_i}$, we conclude $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{c} \circledast \mathbf{i} = \langle \mathbf{c} \mid \mathbf{i}\rangle$ and $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{c}' \circledast (\pi \cdot \mathbf{i}) = \mathbf{c}' \circledast \mathbf{j_i} = \langle \mathbf{c}' \mid \mathbf{j_i}\rangle = \langle \mathbf{c}' \mid \pi \cdot \mathbf{i}\rangle$ by the rewrite rule L.Rewrite. Then, the equation obtained in the previous step becomes

$$\mathcal{E}; \Theta; \Gamma \vdash \exists\, v_i.\ \langle \mathbf{c}' \mid \pi \cdot \mathbf{i}\rangle = \mathbf{shuf\text{-}map}\ (\mathtt{pk}_{\mathbb{CS}}\ sk)\ \langle \mathbf{c} \mid \mathbf{i}\rangle\ v_i. \tag{Φ}$$

As the inputted ciphertext list $\mathbf{c}$ is *well-formed* for the secret key sk by the third hypothesis Eq. ($\mathcal{H}_3$), *i.e.* $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{wf\_ctxt}\ sk\ \mathbf{c}$, we have by the characterisation of the predicate $\mathbf{wf\_ctxt}$ rule L.Wf:Valid, for

all $i \in [\![1; N]\!]$, $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{wf\_ctxt}\ sk\ \langle \mathbf{c} \mid \mathbf{i} \rangle$. Therefore, by the correctness rule for *shuffle-friendly* maps L.Sfm:Correct, and by the equation Eq. (Φ), we have

$$\mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{i=1}^{N} (\mathbf{dec}_{\mathbb{CS}}\ sk\ \langle \mathbf{c}' \mid \pi \cdot \mathbf{i} \rangle = \mathbf{dec}_{\mathbb{CS}}\ sk\ \langle \mathbf{c} \mid \mathbf{i} \rangle).$$

Next, by application of the characterisation rule of $\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}$ L.DecList and by the rewrite rule, the previous equation becomes $\mathcal{E}; \Theta; \Gamma \vdash \bigwedge_{i=1}^{N} (\langle \mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c}' \mid (\pi \cdot \mathbf{i}) \rangle = \langle \mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c} \mid \mathbf{i} \rangle)$. Finally, using this property and because the property $\mathcal{E}; \Theta; \Gamma \vdash \mathbf{perm}_N\ \pi$ holds, the characterisation of multisets equality rule L.Eqm:Charac leads to

$$\boxed{\mathcal{E}; \Theta; \Gamma \vdash \mathbf{eqm}_N\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c}')\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c})}$$

□

D.3.5 Proof of the verifiability property

Now, we can finally prove the verifiability property. We denote by $\mathcal{H}$ and Goal the functions defined by

$$\mathcal{H} \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \lambda r.\ \lambda r'.\mathbf{zkp\text{-}verif}_\pi\ (ck\ n, \mathbf{a}, \mathbf{e})\ \langle \alpha_\pi, r, z_\pi(r) \rangle \wedge \mathbf{zkp\text{-}verif}_\phi\ (ck\ n, \mathbf{a}, \mathtt{pk}_{\mathbb{CS}}\ sk, \mathbf{c}, \mathbf{c}', \mathbf{e}_\phi)\ \langle \alpha_\phi, r', z_\phi(r') \rangle \wedge \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}$$

and

$$\text{Goal} \stackrel{\text{def}}{=} \mathbf{wf\_ctxt}_N\ sk\ \mathbf{c}' \wedge \mathbf{eqm}_N\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c})\ (\mathbf{dec\text{-}list}^{(N)}_{\mathbb{CS}}\ sk\ \mathbf{c}').$$

Hence, the verifiability property consists in proving the following global formula

$$\mathcal{E}; \varnothing \vdash [\mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \text{Goal}].$$

Therefore, by the elimination rule G.LB:Elim of predicate **low-bound** applied to the hypothesis function $\mathcal{H}\ \mathbf{e}_\pi$ and to the goal Goal, we have to prove

$$\mathcal{E}; \varnothing \vdash \tilde{\forall}\, (g' : \mathbf{real}).\ \mathbf{non\text{-}negl}(g')\ \tilde{\wedge}\ \mathbf{det}(g')\ \tilde{\to} [\mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e}_\pi) \to \mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \text{Goal}].$$

Let $g' : \mathbf{real}$ with $\mathbf{non\text{-}negl}(g')$ and $\mathbf{det}(g')$ be a non-negligible deterministic parameter. By another use of the elimination rule G.LB:Elim of predicate **low-bound** applied to the hypothesis function $\mathcal{H}'_{g'} \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e})$ and to the goal $\text{Goal}' \stackrel{\text{def}}{=} \lambda \mathbf{e}.\ \mathcal{H}\ \mathbf{e}\ r_\pi\ r_\phi \to \text{Goal}$, we have to prove

$$\mathcal{E}, (g' : \mathbf{real}); \varnothing \vdash \tilde{\forall}\, (g : \mathbf{real}).\ \mathbf{non\text{-}negl}(g)\ \tilde{\wedge}\ \mathbf{det}(g)\ \tilde{\to} [\mathbf{low\text{-}bound}\ g\ \mathcal{H}'_{g'} \to \mathcal{H}'_{g'}\ \mathbf{e}_\pi \to \text{Goal}'\ \mathbf{e}_\pi].$$

Let $g : \mathbf{real}$ with $\mathbf{non\text{-}negl}(g)$ and $\mathbf{det}(g)$ be a non-negligible deterministic parameter. By putting notations back together, we have to prove the following judgement

$$\mathcal{E}, (g, g' : \mathbf{real}); \varnothing \vdash [\mathbf{low\text{-}bound}\ g\ (\lambda \mathbf{e}.\ \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e})) \to \mathbf{low\text{-}bound}\ g'\ (\mathcal{H}\ \mathbf{e}_\pi) \to \mathcal{H}\ \mathbf{e}_\pi\ r_\pi\ r_\phi \to \text{Goal}]$$

Which is exactly the statement of the last key lemma (Lemma D.5). Therefore, we have achieved the proof of the verifiability property.