A Formal Analysis of
the Norwegian E-Voting Protocol*

Véronique Cortier and Cyrille Wiedling

March 22, 2017

Abstract

Norway used e-voting in its last political election both in September 2011 and September 2013, with more than 28,000 voters using the e-voting option in 2011, and 70,000 in 2013. While some other countries use a black-box, proprietary voting solution, Norway has made its system publicly available. The underlying protocol, designed by Scytl, involves several authorities (a ballot box, a receipt generator, a decryption service, and an auditor). Of course, trusting the correctness and security of e-voting protocols is crucial in that context. In this paper, we propose a formal analysis of the protocol used in Norway, w.r.t. ballot secrecy, considering several corruption scenarios. We use a state-of-the-art definition of ballot secrecy, based on equivalence properties and stated in the applied pi-calculus.

Used in September 2011 and September 2013 for municipality and county elections in Norway [25], E-voting was tested in ten municipalities. During this nationwide local elections, more than 28,000 voters did use internet to cast their vote in 2011 and 70,000 in 2013. While many countries use black-box proprietary solutions, Norway made the protocol publicly available [23]. One key feature of this system is that voters can check that their votes have correctly reached the ballot box ("cast-as-intended" property) without anyone else knowing their vote. The goal of this paper is to conduct a thorough analysis of the Norwegian protocol for the ballot secrecy property: does this system guarantee that no one can know how some voter voted?

Formal methods have been successfully applied to security protocols with the development of several tools such as ProVerif [10], Avispa [4], or Scyther [20] that can automatically analyse both protocols of the literature and fully deployed protocols such as TLS [9]. We therefore chose to model and analyse the Norwegian protocols in a symbolic model named the applied pi-calculus model [1]. A first issue comes from the fact that the underlying encryption primitive is non standard and rather complex. Indeed, the decryption key is split in two shares a_1 and a_2 with $a_3 = a_1 + a_2$. Each of these three keys a_1, a_2, a_3 is given to one administration authority. As further explained in Section 1, this allows for re-encryption and is crucial for the "cast-as-intended" property of the protocol.

Contributions. Our first contribution is a detailed symbolic model for this particular encryption primitive. More precisely, we provide a rewriting system that models El Gamal encryption, re-encryption, blinding function, signatures and zero-knowledge proofs reflecting the primitives used in the protocol. Then, we model the whole system as a process in the applied pi-calculus model. Our second main contribution is a proof of ballot secrecy for several corruption scenarios. Ballot secrecy is typically modeled as follows [21]: an attacker should not be able to distinguish the case where Alice is voting a and Bob is voting b from the converse scenario where Alice is voting b while Bob is voting a. Such a property is typically described by an equivalence of the form

$$\mathsf{Alice}(a) \mid \mathsf{Bob}(b) \approx \mathsf{Alice}(b) \mid \mathsf{Bob}(a)$$

where $\mathsf{Alice}(v)$ represents voter Alice voting for v. Such indistinguishability properties are formalized through behavioral equivalence. Here we use observational equivalence $\approx$ as defined in [1]. Combined with complex equational theories (in particular equational theories with associative and commutative -

---

AC - operators), no existing tool can check for equivalence. Indeed, dedicated tools such as SPEC [34], APTE [15], or AkisS [13] can only handle standard primitives, with the exception of AkisS which covers more primitives but no AC operators. All three tools are moreover limited to a fixed (and small) number of sessions. Recently, the tool Tamarin has been enhanced to cope with equivalence properties [5] but for the moment it requires a high level of interactions. Therefore, the only natural candidate for an automatic analysis of ballot secrecy is the tool ProVerif [10], one of the most generic tools for security protocols. It can check for equivalence [12] for an unbounded number of sessions. However, ProVerif cannot handle equational theories with AC operators either. [2] devises a technique that transforms an equational theory with AC operators into an equational theory without AC operators, that ProVerif can handle. However, [2] is particular to the re-encryption theory (together with any non AC theory) and it is unclear whether it can be adapted to the complex equational theory needed for the Norwegian protocol. Thus we conducted a preliminary analysis of the Norwegian protocol for a simplified - more abstract - model of the protocol (without AC operators), and finite number of voters.

In order to obtain stronger security guarantees, the full equational theory of the cryptographic primitives should be considered. Therefore, the main contribution of the paper is a proof "by hand" of ballot secrecy for two main corruption scenarios: when all authorities are honest (but all but two voters are corrupted) and when the ballot box is corrupted (and again all but two voters are corrupted). We believe that ballot secrecy can be established in a similar way when the receipt generator is corrupted. A preliminary version of ballot secrecy under the first corruption scenario was presented in [18]. While the encryption scheme used in the Norwegian system is particular to the protocol, the Norwegian system also make use of more standard primitives such as signatures or zero-knowledge proofs. When proving ballot secrecy, we developed generic lemmas that could be re-used in subsequent works (e.g [3]).

Related Work. Since our initial study of the Norwegian protocol [18], Gjøsteen has proposed a detailed security analysis of the protocol [24] for both ballot secrecy and verifiability, under several trust assumptions (a first model and security definitions already appeared in [23]). A first main difference between the two approaches is the security model: we consider a symbolic model while [23, 24] rely on a, more concrete, computational model where the attacker is any polynomial probabilistic Turing machine. In that respect, the study in [23, 24] is less abstract and provides more precise security assumptions on the underlying primitives. In contrast, one advantage of symbolic models is that they are more amendable for automation as demonstrated by our use of ProVerif for a simplified model. Ballot secrecy is formalized in two different ways in these two approaches: here, we use a definition of ballot secrecy that is the standard ballot secrecy definition in symbolic models [21]. [23, 24] model ballot secrecy (and verifiability) through an ideal functionality that needs to be adapted to the protocol under consideration. Another difference between [23, 24] and this work lies in the trust assumptions and the properties that are considered. [23, 24] consider both ballot secrecy and verifiability while we focus here on ballot secrecy. Regarding trust assumptions and as it is often the case in other studies of e-voting protocols, at least in symbolic models, we assume that the communications between the voter's computer and the server can be eavesdropped by an attacker, although in practice, these communications typically happen under some secure channel (eg TLS). Whenever possible, it is interesting to prove security without relying on external secure channels since their security is typically not under the control of the authorities, as recently exemplified in an Australian election [26]. In contrast, [23, 24] assume private communications between the voter and the server and consider all the cases where one of the authorities is corrupted (while we do not consider the case of a corrupted receipt generator). Interestingly, [23, 24] shows that the Norwegian protocol remains secure when the decryption device is corrupted while we show this is not the case. This indicates that, when the decryption device is corrupted, ballot secrecy solely relies on the security of the secure channels used by voters.

Several other e-voting protocols have been studied using formal methods. The FOO [22], Okamoto [32], and Lee *et al.* [30] voting protocols have been analysed in [21]. Similarly, Helios has been recently proved secure both in a formal [17] and a computational [7, 8] model. Helios is actually an implementation of a voting system proposed and analyzed (for the available definitions at that time) by Cramer *et al* [19]. All these protocols were significantly simpler to analyse in a symbolic model due to the fact that the cryptographic primitives were easier to abstract as a term algebra and due to the fact that these protocols involve less steps. Civitas has been analyzed in [29] in a symbolic model, for a rather rich equational theory. The

analysis of this protocol remains simpler than the case of the Norwegian protocol, due to the fact that in Civitas, the voting phase does not involve any interaction with the bulletin board. Our study (together with [23, 24]) forms the first security proof of a fully deployed Internet protocol in politically binding elections. Enlarging the scope to voting systems that may take place in polling stations (that is not just Internet voting), security analyses include Scantegrity II [33, 14, 29], Prêt-à-voter [27], and STAR-Vote [6].

Outline of the paper. We provide an informal description of the protocol in Section 1, including details about the different phases of the voting process. Section 3 presents our formal model of the protocol in the applied pi-calculus. The protocol makes use of a special encryption function in combination with signatures, zero-knowledge proofs, blinding functions, and coding functions. We therefore propose a new equational theory reflecting the unusual behavior of the primitives. The main results and corresponding proofs are presented in Section 4. Section 5 gathers the main lemmas needed for the proofs. We believe these lemmas to be of independent interest in the sense they can be useful for further formal studies of different protocols. Most of the proofs are detailed in Sections F and 6 with some additional lemmas postponed to the appendix. Finally, in Section 7, we present a simplified model of the Norwegian protocol and the corresponding security analysis using ProVerif.

1 The Norwegian E-Voting Protocol

The Norwegian protocol features four players that define the electronic poll's infrastructure: a Ballot box (B), a Receipt generator (R), a Decryption service (D) and an Auditor (A). Each Voter (V) can log in using a Computer (P) in order to submit his vote. Channels between computers (voters) and the Ballot box are considered to be authenticated channels, channels between infrastructure's player are untappable, and channels between voters and receipt generator are unidirectional out-of-band channels. (Example of SMS is given in [23].) The protocol can be divided in three phases: the setting phase, the submission phase, where voters submit their votes, and the counting phase, where ballots are counted and the auditor verifies the correctness of the election.

1.1 Setting Phase

Before the election, a finite cyclic group G of some prime order q and generator by g is selected, three private keys a_1, a_2, and a_3 (such that $\mathsf{a}_1 + \mathsf{a}_2 \equiv \mathsf{a}_3\ [q]$) are generated and distributed over respectively D, B, and R, while the corresponding public keys $\mathsf{y}_1 = \mathsf{g}^{\mathsf{a}_1}$, $\mathsf{y}_2 = \mathsf{g}^{\mathsf{a}_2}$ and $\mathsf{y}_3 = \mathsf{g}^{\mathsf{a}_3}$ are made publicly available. Each voter V is assumed to have a signing key id_V, with a corresponding verification key, $\mathsf{vk}(\mathsf{id}_V)$, which is public. The Ballot box B is provided with a table $V \mapsto \mathsf{s}_V$ that associates each voter with a blinding factor s_V. The Receipt generator R is also assumed to have a signing key id_R, with a corresponding public verification key, $\mathsf{vk}(\mathsf{id}_R)$; and it is given, for each voter V, a pseudo-random function $d_V : G \to \mathcal{C}$, where $\mathcal{C}$ is the set of receipts. Finally, each voter V is assumed to receive by surface mail a table that associates to any voting option o (from the set $\mathcal{O}$), a precomputed receipt code $d_V(\mathsf{f}(\mathsf{o})^{\mathsf{s}_V}) \in \mathcal{C}$, where $\mathsf{f} : \mathcal{O} \to G$ is an injective encoding function.

1.2 Submission Phase

The submission phase is summarized in Figure 1. We detail in this section the expected behavior of each participant.

Voter (V). Each voter tells his computer what voting option o to submit and allows it to sign the corresponding ballot on his behalf. Then, he has to wait for an acceptance message coming from the computer and a receipt $\check{\mathsf{r}}$ sent by the receipt generator through some out-of-band channel (typically a SMS message). Using the receipt, he verifies that the correct vote was submitted, that is, he checks that $\check{\mathsf{r}} = d_V(\mathsf{f}(\mathsf{o})^{\mathsf{s}_V})$ by verifying that the receipt code $\check{\mathsf{r}}$ indeed appears in the line associated to the voting option o he has chosen.

This check ensures in particular the "cast-as-intended" property. In case the voter's computer is corrupted and encrypts another vote (say the computer wishes to cast a vote for the Pirate party) then this would be eventually discovered by the voter when receiving the receipt.
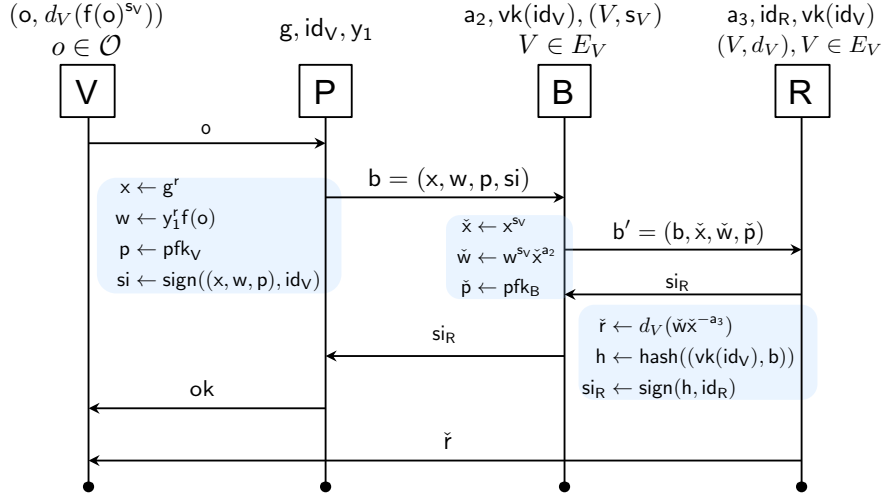
Figure 1: Submission of one vote.

Computer (P). Voter's computer encrypts voter's ballot with the public key y_1 using standard El Gamal encryption. The resulting ballot is $\langle g^r, y_1^r f(o)\rangle$. P also proves that the resulting ciphertext corresponds to a valid voting option, by computing a standard proof of knowledge $\mathsf{pfk_V}$. (This proof is formally presented in Section 3, but a detailed description can be found in [23].). P also signs, on the behalf of the Voter, the ballot with $\mathsf{id_V}$ and sends it to the Ballot box. It then waits for a confirmation $\mathsf{si_R}$ coming from the latter, which is a hash of the initial encrypted ballot, signed by the Receipt generator. After checking this signature, the computer notifies the voter that his vote has been taken into account.

Ballot box (B). Upon receiving an encrypted and signed ballot b from a computer, the Ballot box first checks the correctness of signatures and proofs before re-encrypting the original encrypted ballot with $\mathsf{a_2}$ and blinding it with $\mathsf{s_V}$. B also generates a proof $\mathsf{pfk_B}$, showing that its computation is correct. B then sends the new modified ballot $\mathsf{b'}$ to the Receipt generator. Once the Ballot box receives a message $\mathsf{si_R}$ from R, it simply checks that the Receipt generator's signature is valid, and sends it to the computer.

Receipt generator (R). Upon receiving an encrypted ballot $\mathsf{b'} = \langle \mathsf{b}, \check{\mathsf{x}}, \check{\mathsf{w}}, \check{\mathsf{p}}\rangle$ from the Ballot box, the Receipt generator first checks signature and proofs (from the computer and the Ballot box). If the validity checks are successful, it generates:

- a receipt code $\check{\mathsf{r}} = d_V(\check{\mathsf{w}}\check{\mathsf{x}}^{-\mathsf{a_3}})$ sent by out-of-band channel directly to the Voter. Intuitively, the Receipt generator decrypts the (blinded) ballot, applying the function d_V associated to the voter. This receipt code gives assurance to the voter that the correct vote was submitted to the Ballot box.

- a signature on a hash of the original encrypted ballot for the Ballot box. Once transmitted by B, it is checked and passed on to the Voter's Computer, which checks it once more and informs the Voter that his vote has been taken into account.

1.3 Counting Phase

Once the voting phase is over, the counting phase begins (Figure 2). The Ballot box selects the encrypted votes $x_1, \ldots, x_k$ which need to be decrypted (if a voter has voted several times, all the submitted ballots remain in the memory of the Ballot box but only the last ballot should be sent) and sends them to the Decryption service. The whole content of the Ballot box $\mathsf{b_1}, \ldots, \mathsf{b_n}$ $(n \geq k)$ is revealed to the Auditor, including previous votes from re-voting voters. The Receipt generator sends to the Auditor the list of hashes of ballots it has seen during the submission phase. The Decryption service decrypts the incoming
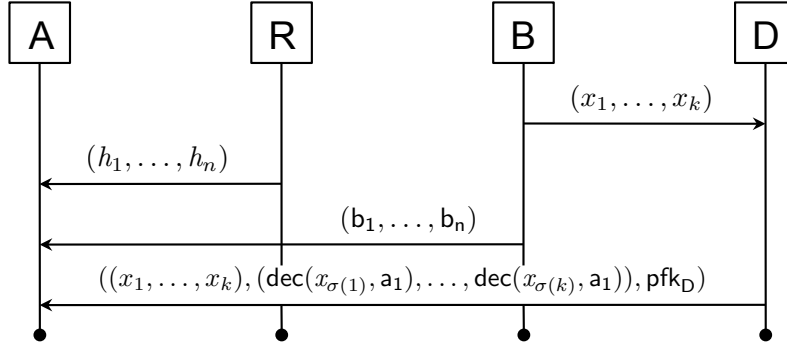
Figure 2: Counting phase.

ciphertexts $x_1, \ldots, x_k$ received from the Ballot box and shuffles the decrypted votes before publishing them. It therefore outputs a message of the form $\mathsf{dec}(x_{\sigma(1)}, \mathsf{a}_1), \ldots, \mathsf{dec}(x_{\sigma(k)}, \mathsf{a}_1)$ where σ denotes the permutation obtained by shuffling the votes. It also provides the Auditor with a proof pfk_D showing that the input ciphertexts and the outcoming decryption indeed match. Using the Ballot box content and the list of hashes from the Receipt generator, the Auditor verifies that no ballots have been inserted or lost and it computes its own list of encrypted ballots which should be counted. He compares this list with the one received from the Decryption service and checks the proof provided by the latter.

1.4 Security Analysis

The rest of the paper is devoted to the modeling and analysis of the protocol. We summarize here informally our main results and we list our main assumptions and simplifications. We formally prove ballot secrecy under two threat scenario:

- All but two voters are dishonest and all the authorities (the Ballot box, the Receipt generator, the Decryption service and the Auditor) are honest.

- All but two voters are dishonest and all the authorities but the Ballot box are honest.

These two results are obtained assuming authenticated but not necessarily secure communication channels between voters and the Ballot box and Receipt generator, meaning that an attacker may eavesdrop the communications. For this reason, ballot secrecy does not hold as soon as the Decryption device is corrupted. Note also that ballot privacy is also broken as soon as the voter's computer (P) is corrupted since the voter sends her vote to her computer. We therefore had to assume P to be honest and we chose to model it together with the voter's behavior. Authentication between the voter and the Ballot box is ensured through a login and password mechanism while authentication from the receipt generator to the voter relies on the out-of-band channel used between them (typically a SMS message).

When modeling the Norwegian protocol in a symbolic model, we had to proceed to some simplifications. First, and as discussed later in Section 3.1, the cryptographic primitives are abstracted by terms together with an equational theory, which potentially leaves out some flaws due to some crafty modifications of some messages. Second, we chose not to model revoting, for simplicity but also since it is explicitly and strongly discouraged in [23], as it may provide to an attacker that control the Ballot box the opportunity to swap the initial and the resubmitted votes. Third, for simplicity, we only consider one-option votes (that voters select one option), rather that a vector of options as in [23]. Finally, we also omit the proof of correct decryption provided by the decryption device since it should not affect ballot secrecy.

2 Applied Pi-Calculus

We describe here the applied pi-calculus [1], introduced by M. Abadi et C. Fournet. The applied pi-calculus is a process algebra that is often used to model protocols, and we briefly recall here the notations

and definitions, providing some examples.

2.1 Terms

Messages are represented by *terms* built upon an infinite set of *names* $\mathcal{N}$ (for communication channels or atomic data), a set of *variables* $\mathcal{X}$ and a *signature* Σ consisting of a finite set of *function symbols* (to represent cryptographic primitives). A function symbol f is assumed to be given with its arity $\mathsf{ar}(f)$. Then, the set of terms $T(\Sigma, \mathcal{X}, \mathcal{N})$ is formally defined by the following grammar:

$$
\begin{aligned}
t, t_1, t_2, \ldots ::= & \\
& x & & x \in \mathcal{X} \\
& n & & n \in \mathcal{N} \\
& f(t_1, \ldots, t_n) & & f \in \Sigma, \; n = \mathsf{ar}(f)
\end{aligned}
$$

We write $\{^{M_1}/_{x_1}, \ldots, ^{M_n}/_{x_n}\}$ for the *substitution* that replaces the variables x_i with the terms M_i. $N\sigma$ refers to the result of applying substitution σ to the free variables of the term N. A term is called *ground* when it does not contain variables.

In order to represent the properties of the primitives, the signature Σ is equipped with an *equational theory* E that is a set of equations which hold on terms built from the signature. We denote by $=_E$ the smallest equivalence relation induced by E, closed under application of function symbols, substitutions of terms for variables and bijective renaming of names. We write $M =_E N$ when the equation $M = N$ holds in the theory E.

Example 1. *A signature for symmetric, asymmetric encryption and signature is*

$$\Sigma = \{\mathsf{checksign}, \mathsf{dec}, \mathsf{pdec}, \mathsf{enc}, \mathsf{penc}, \mathsf{sign}, \mathsf{pk}\}$$

where penc *and* pdec *represent resp. asymmetric (randomized) encryption and decryption,* enc *and* dec *stand resp. for symmetric (randomized) encryption and decryption,* sign *models signature,* $\mathsf{checksign}$ *represents a function that checks the validity of signature, and* pk *represents the public key associated to a secret key. For example, the term* $\mathsf{penc}(m, r, \mathsf{pk}(sk_a))$ *represents the asymmetric encryption of the message* m *with the public key corresponding to the secret key* sk_a *and random factor* r. *The properties of the cryptographic functions are represented by the equational theory* E_{enc}:

$$
\begin{aligned}
\mathsf{dec}(\mathsf{enc}(m, r, k), k) &= m \\
\mathsf{pdec}(\mathsf{penc}(m, r, \mathsf{pk}(sk)), sk) &= m \\
\mathsf{checksign}(\mathsf{sign}(m, sk), \mathsf{pk}(sk), m) &= \mathsf{ok}.
\end{aligned}
$$

The first equation models that symmetric decryption is only successful if the key used for decryption is the same as the one used for encryption. The second equation reflects that asymmetric decryption only succeeds when the corresponding secret key is used. Finally, the last equation checks that a signature corresponds to a given message and a given verification key.

For some cryptographic primitives, such as homomorphic encryption, it is necessary to introduce associative and commutative symbols. Equational theories including such symbols are called *AC-theories*. Therefore we define an equality modulo AC, noted $=_{AC}$, which denotes that two terms are syntactically equal modulo the associative or commutative properties for each AC symbol $+$.

$$
\begin{aligned}
x + (y + z) &= (x + y) + z \\
x + y &= y + x
\end{aligned}
$$

Example 2. *To represent homomorphic encryption, we may use the signature* $\Sigma = \{+, *, \mathsf{enc}\}$ *with* $+$ *and* $*$ *two associative and commutative (AC) symbols and the corresponding equational theory* E_{AC}, *which includes the associative and commutative properties of* $+$ *and* $*$ *symbols, and the equation:*

$$\mathsf{enc}(m, k) * \mathsf{enc}(n, k) = \mathsf{enc}(m + n, k).$$

This equation models that two ciphertexts encrypted with the same key can be combined to create a ciphertext which corresponding plaintext is the sum of the two previous plaintexts.

$$
\begin{array}{lll}
P, Q, R ::= & & \text{(plain) processes} \\
\quad 0 & & \text{null process} \\
\quad P \mid Q & & \text{parallel composition} \\
\quad !P & & \text{replication} \\
\quad \nu n.P & & \text{name restriction} \\
\quad \text{if } \phi \text{ then } P \text{ else } Q & & \text{conditional} \\
\quad u(x).P & & \text{message input} \\
\quad \overline{u}\langle M \rangle.P & & \text{message output} \\
& & \\
A, B, C ::= & & \text{extended processes} \\
\quad P & & \text{plain process} \\
\quad A \mid B & & \text{parallel composition} \\
\quad \nu n.A & & \text{name restriction} \\
\quad \nu x.A & & \text{variable restriction} \\
\quad \{^{M}/_{x}\} & & \text{active substitution}
\end{array}
$$

Figure 3: Syntax for processes

2.2 Rewriting System

It might be difficult to work modulo an equational theory. Instead, it is often possible (and more convenient) to reason with a rewriting system. Formally, a *rewriting system* $\mathcal{R}$ is a set of *rewriting rules* of the form $l \to r$ with l and r two terms. We say that a term s is *rewritten* in t for rule $l \to r$, noted $s \to t$, if there exists a position p in s and a substitution θ such that $s|_p = l\theta$ and $t = s[r\theta]_p$.

Definition 1 (Convergence). *A rewriting system $\mathcal{R}$ is said* convergent *if:*

- *for every ground term U, there exists no infinite sequence $U \to U_1 \to \cdots \to U_k \to \ldots$.*
  (In this case, we say that $\mathcal{R}$ is terminating.)

- *for every ground terms U, U_1, and U_2 such that $U \to^* U_1$ and $U \to^* U_2$, there exists V such that $U_1 \to^* V$, and $U_2 \to V$. (In this case, we say that $\mathcal{R}$ is confluent.)*

ForAC-theories, we consider rewriting systems modulo AC. Formally, a term s is rewritten modulo AC in t, noted $s \to_{AC} t$, for a rule $l \to r$ if there exist s' and t' two terms such that $s =_{AC} s'$, $t =_{AC} t'$ and s' is rewritten in t' for the rule $l \to r$. Then, it is possible to define the notion of AC-convergence.

Definition 2 (AC-Convergence). *A rewriting system $\mathcal{R}$ is said AC-convergent if:*

- *for every ground term U, there is no infinite sequence $U \to_{AC} U_1 \to_{AC} \cdots \to_{AC} U_k \ldots$.*
  (In this case, we say that $\mathcal{R}$ is AC-terminating.)

- *for every ground terms U, U_1, and U_2 such that $U \to^*_{AC} U_1$ and $U \to^*_{AC} U_2$, there exists V such that $U_1 \to^*_{AC} V$, and $U_2 \to^*_{AC} V$. (In this case, we say that $\mathcal{R}$ is AC-confluent.)*

2.3 Processes

Processes and *extended processes* are defined in Figure 3. The process 0 represents the null process that does nothing. $P \mid Q$ denotes the parallel composition of P with Q while $!P$ denotes the unbounded replication of P (*i.e.* the unbounded parallel composition of P with itself). $\nu n.P$ creates a fresh name n and then behaves like P. The process if ϕ then P else Q behaves like P if ϕ holds and like Q otherwise. $u(x).P$ inputs some message in the variable x on channel u and then behaves like P while $\overline{u}\langle M \rangle.P$ outputs M on channel u and then behaves like P. We write $\nu \tilde{u}$ for the (possibly empty) series of pairwise-distinct binders $\nu u_1 . \ldots . \nu u_n$. The active substitution $\{^{M}/_{x}\}$ can replace the variable x for the term M in every process it comes into contact with and this behaviour can be controlled by restriction, in particular, the process $\nu x \left(\{^{M}/_{x}\} \mid P \right)$ corresponds exactly to let $x = M$ in P.

$$
\begin{array}{llrcl}
\text{PAR} - 0 & & A & \equiv & A \mid 0 \\
\text{PAR-A} & & A \mid (B \mid C) & \equiv & (A \mid B) \mid C \\
\text{PAR-C} & & A \mid B & \equiv & B \mid A \\
\text{REPL} & & !P & \equiv & P \mid !P \\
\text{NEW} - 0 & & \nu n.0 & \equiv & 0 \\
\text{NEW-C} & & \nu u.\nu w.A & \equiv & \nu w.\nu u.A \\
\text{NEW-PAR} & & A \mid \nu u.B & \equiv & \nu u.(A \mid B) \qquad \text{if } u \notin \mathsf{fv}(A) \cup \mathsf{fn}(A) \\
\text{ALIAS} & & \nu x.\{^M/_x\} & \equiv & 0 \\
\text{SUBST} & & \{^M/_x\} \mid A & \equiv & \{^M/_x\} \mid A\{^M/_x\} \\
\text{REWRITE} & & \{^M/_x\} & \equiv & \{^N/_x\} \qquad \text{if } M =_E N
\end{array}
$$

Figure 4: Structural equivalence.

As in [17], we slightly extend the applied pi-calculus by letting conditional branches now depend on formulae defined by the following grammar:

$$\phi, \psi ::= M = N \mid M \neq N \mid \phi \wedge \psi$$

If M and N are ground, we define $[\![M = N]\!]$ to be true if $M =_E N$ and false otherwise. The semantics of $[\![\ ]\!]$ is then extended to formulae as expected.

The *scope* of names and variables is delimited by binders $u(x)$ and νu. Sets of bound names, bound variables, free names and free variables are respectively written $\mathsf{bn}(A)$, $\mathsf{bv}(A)$, $\mathsf{fn}(A)$ and $\mathsf{fv}(A)$. Occasionally, we write $\mathsf{fn}(M)$ (resp. $\mathsf{fv}(M)$) for the set of names (resp. variables) which appear in term M. An extended process is *closed* if all its variables are either bound or defined by an active substitution.

An *context* $C[\_]$ is an extended process with a hole instead of an extended process. We obtain $C[A]$ as the result of filling $C[\_]$'s hole with the extended process A. An *evaluation context* is a context whose hole is not in the scope of a replication, a conditional, an input or an output. A context $C[\_]$ closes A when $C[A]$ is closed.

A *frame* is an extended process built up from the null process 0 and active substitutions composed by parallel composition and restriction. The *domain* of a frame φ, denoted $\mathsf{dom}(\varphi)$ is the set of variables for which φ contains an active substitution $\{^M/_x\}$ such that x is not under restriction. Every extended process A can be mapped to a frame $\varphi(A)$ by replacing every plain process in A with 0.

We refer the reader to Section 2.6 for a full example.

2.4 Operational Semantics

The operational semantics of processes in the applied pi-calculus is defined by three relations: *structural equivalence* ($\equiv$), *internal reduction* ($\rightarrow$) and *labelled reduction* ($\xrightarrow{\alpha}$).

Structural equivalence is defined in Figure 4. It is closed by α-conversion of both bound names and bound variables, and closed under application of evaluation contexts. Structural equivalence corresponds to some structural rewriting that does not change the semantics of a process. The *internal reductions* and *labelled reductions* are defined in Figure 5. They are closed under structural equivalence and application of evaluation contexts. Internal reductions represent evaluation of condition and internal communication between processes. Labelled reductions represent communications with the environment.

2.5 Equivalences

Privacy properties are often stated as equivalence relations [21]. Intuitively, if a protocol preserves ballot secrecy, an attacker should not be able to distinguish between a scenario where a voter votes 0 from a scenario where the voter votes 1. *Static equivalence* formally expresses the indistinguishability of two sequences of terms.

Definition 3 (Static equivalence). *Two closed frames φ and ψ are statically equivalent, denoted $\varphi \approx_s \psi$, if $\mathsf{dom}(\varphi) = \mathsf{dom}(\psi)$ and there exists a set of names $\tilde{n}$ and substitutions σ, τ such that $\varphi \equiv \nu \tilde{n}.\sigma$ and*

$$\text{(COMM)} \quad \overline{c}\langle M\rangle.P \mid c(x).Q \to P \mid Q\{^M/_x\}$$

$$\text{(THEN)} \quad \text{if } \phi \text{ then } P \text{ else } Q \to P \quad \text{if } \llbracket \phi \rrbracket = \text{true}$$

$$\text{(ELSE)} \quad \text{if } \phi \text{ then } P \text{ else } Q \to Q \quad \text{otherwise}$$

$$\text{(IN)} \qquad c(x).P \xrightarrow{c(M)} P\{^M/_x\}$$

$$\text{(OUT-ATOM)} \qquad \overline{c}\langle u\rangle.P \xrightarrow{\overline{c}\langle u\rangle} P$$

$$\text{(OPEN-ATOM)} \qquad \frac{A \xrightarrow{\overline{c}\langle u\rangle} A' \qquad u \neq c}{\nu u.A \xrightarrow{\nu u.\overline{c}\langle u\rangle} A'}$$

$$\text{(SCOPE)} \qquad \frac{A \xrightarrow{\alpha} A' \qquad u \text{ does not occur in } \alpha}{\nu u.A \xrightarrow{\alpha} \nu u.A'}$$

$$\text{(PAR)} \qquad \frac{A \xrightarrow{\alpha} A' \qquad \mathsf{bv}(\alpha) \cap \mathsf{fv}(B) = \mathsf{bn}(\alpha) \cap \mathsf{fn}(B) = \emptyset}{A \mid B \xrightarrow{\alpha} A' \mid B}$$

$$\text{(STRUCT)} \qquad \frac{A \equiv B \qquad B \xrightarrow{\alpha} B' \qquad B' \equiv A'}{A \xrightarrow{\alpha} A'}$$

where α is a *label* of the form $c(M)$, $\overline{c}\langle u\rangle$, or $\nu u.\overline{c}\langle u\rangle$
such that u is either a channel name or a variable of base type.

Figure 5: Semantics for processes

$\psi \equiv \nu\tilde{n}.\tau$ *and for all terms M, N such that $\tilde{n} \cap (\mathsf{fn}(M) \cup \mathsf{fn}(N)) = \emptyset$, we have $M\sigma =_E N\sigma$ holds if and only if $M\tau =_E N\tau$ holds.*

Two closed extended processes A, B are statically equivalent, written $A \approx_s B$, if their frames are statically equivalent; that is, $\varphi(A) \approx_s \varphi(B)$.

Intuitively, two sequences of messages φ and ψ are distinguishable to an attacker (*i.e.* they are not statically equivalent) if the attacker can build a public test $M = N$ that holds for φ but not for ψ (or the converse).

Example 3. *Consider the signature and equational theory E_{enc} defined in Example 1. Let $\varphi_1 = \nu k.\sigma_1$ and $\varphi_2 = \nu k.\sigma_2$ where $\sigma_1 = \{^{\mathsf{penc}(s_1,r_1,\mathsf{pk}(k))}/_{x_1}, \ ^{\mathsf{pk}(k)}/_{x_2}\}$, $\sigma_2 = \{^{\mathsf{penc}(s_2,r_2,\mathsf{pk}(k))}/_{x_1}, \ ^{\mathsf{pk}(k)}/_{x_2}\}$ and s_1, s_2, k are names. We have that $\varphi_1 \not\approx_s \varphi_2$. Indeed, we have $\mathsf{penc}(s_1,r_1,x_2)\sigma_1 =_E x_1\sigma_1$ but $\mathsf{penc}(s_1,r_1,x_2)\sigma_2 \neq_E x_1\sigma_2$.*

Intuitively, since the randomness of the encryption is public, an attacker may reconstruct the ciphertexts and compare. The two messages become indistinguishable as soon as the randomness remain private. That is, we have that $\nu(k,r_1).\sigma_1 \approx_s \nu(k,r_2).\sigma_2$.

Observational equivalence is the active counterpart of static equivalence, where the attacker can actively interact with the processes. The definition of observational equivalence requires to reason about all contexts (*i.e.* all adversaries), which renders the proofs difficult. Since observational equivalence has been shown to coincide [1, 31] with labelled bisimilarity, we adopt the latter in the remaining of the paper.

Definition 4 (Labelled bisimilarity). *Labelled bisimilarity ($\approx_l$) is the largest symmetric relation $\mathcal{R}$ on closed extended processes such that $A\mathcal{R}B$ implies:*

1. *$A \approx_s B$;*

2. *if $A \to A'$, then $B \to^* B'$ and $A'\mathcal{R}B'$ for some B';*

3. *if $A \xrightarrow{\alpha} A'$ such that $\mathsf{fv}(\alpha) \subseteq \mathsf{dom}(A)$ and $\mathsf{bn}(\alpha) \cap \mathsf{fn}(B) = \emptyset$, then $B \to^* \xrightarrow{\alpha} \to^* B'$ and $A'\mathcal{R}B'$ for some B'.*

Intuitively, two processes A and B are labelled bisimilar if, anyhow the process A (resp. B) behaves, the process B (resp. A) may behave with the same visible actions and such that their resulting frames are statically equivalent, that is, an attacker cannot distinguish between them.

2.6 A Detailed Example

We provide a detailed example to illustrate the syntax and semantics of the applied pi-calculus. Readers already familiar with this formalism may skip this section. Let us consider a simple protocol proposed (for illustration purposes) by B. Blanchet [11].

$$\text{Alice} \quad \xrightarrow{\{\mathsf{sign}(k,\mathsf{sk_A})\}_{\mathsf{pk_B}}} \quad \text{Bob}$$
$$\xleftarrow{\{s\}_k}$$

In this protocol, Alice sends a newly generated and signed key k to Bob, the whole message encrypted using his public key $\mathsf{pk_B}$. Then, the receiver (Bob) checks that the signature belongs to the intended sender (Alice). He then encrypts a fresh secret s with k using symmetric encryption and sends it to Alice.

To model this protocol in applied-pi calculus, we use the equational theory E_{enc} described in Example 1 that models the properties of the different primitives used in the protocol. The behavior of the sender and receiver are modeled in the applied pi-calculus as follows.

$$A := \nu k \,.\, \nu r_a \,.\, \bar{c}\langle \mathsf{penc}(\mathsf{sign}(k, sk_a), r_a, \mathsf{pk_B}) \rangle \,.\, c(x_1)$$
$$B(s) := c(x_2) \,.\, B'(s)$$
$$B'(s) := \text{let } y = \mathsf{pdec}(x_2, sk_b) \text{ in}$$
$$\text{if } \mathsf{checksign}(y, \mathsf{pk_A}) = \mathsf{ok} \text{ then } B''(s)$$
$$B''(s) := \nu r_b \,.\, \bar{c}\langle \mathsf{enc}(s, r_b, k) \rangle$$

In these processes, a and b are secret values that correspond to the private key pairs of Alice and Bob, respectively. Since Alice should not have a direct access to b, the secret key of Bob, A is given access to Bob's public key through a variable $\mathsf{pk_B}$ and similarly for B. The whole protocol can be easily expressed using Alice and Bob processes:

$$P(s) := \nu(a, b) \,.\, [A \mid B(s) \mid \Gamma]$$

where $\Gamma = \left\{ {}^{\mathsf{pk}(sk_a)}/_{\mathsf{pk_B}}, {}^{\mathsf{pk}(sk_b)}/_{\mathsf{pk_B}} \right\}$. As mentioned above, sk_a and sk_b are protected but since $\mathsf{pk}(sk_a)$ and $\mathsf{pk}(sk_b)$ are public keys, we published them in a frame implying that they are available to anyone (including the attacker). We first illustrate the semantics of the internal reductions by simulating the normal execution of the protocol, without any interference.

$$P(s) \xrightarrow{\text{(COMM)}} \nu(sk_a, sk_b, k, r_a). \left[c(x_1) \mid B'(s) \mid \left\{ {}^{\mathsf{penc}(\mathsf{sign}(k,sk_a),r_a,\mathsf{pk_B})}/_{x_2} \right\} \mid \Gamma \right]$$
$$\xrightarrow{\text{(THEN)}} \nu(sk_a, sk_b, k, r_a). \left[c(x_1) \mid B''(s) \mid \left\{ {}^{\mathsf{penc}(\mathsf{sign}(k,sk_a),r_a,\mathsf{pk_B})}/_{x_2} \right\} \mid \Gamma \right]$$
$$\xrightarrow{\text{(COMM)}} \nu\tilde{n}. \left[\left\{ {}^{\mathsf{enc}(s,r_b,k)}/_{x_1}, {}^{\mathsf{penc}(\mathsf{sign}(k,sk_a),r_a,\mathsf{pk_B})}/_{x_2} \right\} \mid \Gamma \right], \text{ with } \tilde{n} = (sk_a, sk_b, k, r_a, r_b).$$

Alice sends her signed and encrypted message, which is received by Bob.

Actually, this simple protocol is flawed. Indeed, an intruder may impersonate Alice's identity from Bob's point of view. This attack only requires that Alice had, once in the past, spoken to the intruder using this protocol. This can be showed using our applied-pi calculus model by adding $\mathsf{penc}(\mathsf{sign}(k_c, sk_a), r_c, \mathsf{pk_C})$ to the initial frame, which is a message that Alice would have sent to Charlie according to the protocol. Let us define this slightly new process:

$$P_c(s) := \nu(sk_a, sk_b, sk_c, r_c, k_c) \,.\, [A \mid B(s) \mid \Gamma_c],$$

where $\Gamma_c = \left\{ {}^{\mathsf{penc}(\mathsf{sign}(k_c,sk_a),r_c,\mathsf{pk_C})}/_x, {}^{\mathsf{pk}(a)}/_{\mathsf{pk_A}}, {}^{\mathsf{pk}(b)}/_{\mathsf{pk_B}}, {}^{\mathsf{pk}(sk_c)}/_{\mathsf{pk_C}} \right\}$. Let us consider now the following execution: Charlie can send a message $M = \mathsf{penc}(\mathsf{pdec}(x, sk_C), r', \mathsf{pk_B})$ to Bob, where x is the message

previously received from Alice.

$$P_c(s) \xrightarrow{c(M)} \nu(sk_a, sk_b, r_c, k_c).\,[A \mid B' \mid \Gamma_c]$$

$$\xrightarrow{\text{(THEN)}} \nu(sk_a, sk_b, r_c, k_c).\,[A \mid B'' \mid \Gamma_c]$$

$$\xrightarrow{\nu x_c.\bar{c}\langle x_c\rangle} \nu(sk_a, sk_b, r_c, k_c, r_b).\,\left[A \mid \left\{^{\mathsf{enc}(s,r_b,k_c)}/_{x_c}\right\} \mid \Gamma_c\right].$$

Let us define the substitution $\sigma = \left\{^{\mathsf{enc}(s,r_b,k_c)}/_{x_c},{}^{M}/_{x_2}\right\} \mid \Gamma_c$. In this execution, the test performed by B succeeds since, according to the equation theory, we have:

$$M\sigma = \mathsf{penc}(\mathsf{pdec}(x, \mathsf{sk}(c)), r', \mathsf{pk_B})\sigma$$
$$= \mathsf{penc}(\mathsf{pdec}(\mathsf{penc}(\mathsf{sign}(k_c, \mathsf{sk}(a)), r_c, \mathsf{pk}(c)), \mathsf{sk}(c)), r', \mathsf{pk}(b))$$
$$=_{E_{\mathsf{enc}}} \mathsf{penc}(\mathsf{sign}(k_c, \mathsf{sk}(a)), r', \mathsf{pk}(b)),$$

thus,

$$\mathsf{checksign}(\mathsf{pdec}(x_2, \mathsf{sk}(b)), \mathsf{pk_A})\sigma$$
$$=_{E_{\mathsf{enc}}} \mathsf{checksign}(\mathsf{pdec}(\mathsf{penc}(\mathsf{sign}(k_c, \mathsf{sk}(a)), r', \mathsf{pk}(b)), \mathsf{sk}(b)), \mathsf{pk}(a))$$
$$=_{E_{\mathsf{enc}}} \mathsf{checksign}(\mathsf{sign}(k_c, \mathsf{sk}(a)), \mathsf{pk}(a))$$
$$=_{E_{\mathsf{enc}}} \mathsf{ok}$$

which implies that the message is accepted by B, who therefore believes that Alice just sent him the key k_c she had, in fact, sent to Charlie in a former session. At the end of this execution, one can see that s is not secret anymore since Charlie knows k_c from his previous exchange and can perform a decryption of x_c to get s.

We can also illustrate Definition 4 using this example. Indeed, we have that for any s_1 and s_2, $P(s_1) \not\approx_l P(s_2)$. This is due to the fact that the two processes may evolve in two states that are not statically equivalent ($\approx_s$):

$$P(s_1) \xrightarrow{\alpha}{}^* P_1 = \nu(a, b, r_b).\,[A \mid \sigma_1] \text{ and } P(s_2) \xrightarrow{\alpha}{}^* P_2 = \nu(a, b, r_b).\,[A \mid \sigma_2]$$

where $\alpha = c(M).\nu x_c.\bar{c}\langle x_c\rangle.$ and σ_i is defined as σ where s is simply replaced by s_i. We can show that $P_1 \not\approx_s P_2$, due to the fact that the intruder can observe, using $N_1 = \mathsf{dec}(x_c, k_c)$ and $N_2 = s_1$, that $(N_1 =_E N_2)\sigma_1$ but $(N_1 \neq_E N_2)\sigma_2$.

3 Modeling the Norwegian Protocol

We provide a formal specification of the Norwegian protocol, using the framework of the applied pi-calculus, defined in the previous section. We first model the cryptographic primitives used in the protocol (Section 3.1) and then the Norwegian protocol itself (Section 3.2).

3.1 Equational Theory

We adopt the following signature to capture the cryptographic primitives used by the protocol.

$$\Sigma_{\mathsf{sign}} = \{\mathsf{ok}, \mathsf{fst}, \mathsf{hash}, \mathsf{p}, \mathsf{pk}, \mathsf{s}, \mathsf{snd}, \mathsf{vk}, \mathsf{blind}, \mathsf{d}, \mathsf{dec}, +, *, \circ, \diamond, \mathsf{pair},$$
$$\mathsf{renc}, \mathsf{sign}, \mathsf{unblind}, \mathsf{checkpfk}_1, \mathsf{checkpfk}_2, \mathsf{checksign}, \mathsf{penc}, \mathsf{pfk}_1, \mathsf{pfk}_2\}$$

The function ok is a constant; $\mathsf{fst}, \mathsf{hash}, \mathsf{p}, \mathsf{pk}, \mathsf{s}, \mathsf{snd}, \mathsf{vk}$ are unary functions; $\mathsf{blind}, \mathsf{d}, \mathsf{dec}, +, *, \circ, \diamond, \mathsf{pair}, \mathsf{renc}, \mathsf{sign}, \mathsf{unblind}$ are binary functions; $\mathsf{checkpfk}_1, \mathsf{checksign}, \mathsf{penc}$ are ternary functions; $\mathsf{pfk}_1, \mathsf{checkpfk}_2$ are quaternary functions and pfk_2 is a quinary function.

The term $\mathsf{pk}(K)$ denotes the public key corresponding to the secret key K in asymmetric encryption. Terms $\mathsf{s}(I)$, $\mathsf{p}(I)$, and $\mathsf{vk}(I)$ are respectively the blinding factor, the parameter and the verification key

$$\mathsf{fst}(\mathsf{pair}(x, y)) = x \qquad\qquad\text{(E-1)}$$

$$\mathsf{snd}(\mathsf{pair}(x, y)) = y \qquad\qquad\text{(E-2)}$$

$$\mathsf{dec}(\mathsf{penc}(x, r, \mathsf{pk}(k)), k) = x \qquad\qquad\text{(E-3)}$$

$$\mathsf{dec}(\mathsf{blind}(\mathsf{penc}(x, r, \mathsf{pk}(k)), b), k) = \mathsf{blind}(x, b) \qquad\qquad\text{(E-4)}$$

$$\mathsf{penc}(x, r_1, k_p) \circ \mathsf{penc}(y, r_2, k_p) = \mathsf{penc}(x \diamond y, r_1 * r_2, k_p) \qquad\qquad\text{(E-5)}$$

$$\mathsf{renc}(\mathsf{penc}(x, r, \mathsf{pk}(k_1)), k_2) = \mathsf{penc}(x, r, \mathsf{pk}(k_1 + k_2)) \qquad\qquad\text{(E-6)}$$

$$\mathsf{unblind}(\mathsf{blind}(x, b), b) = x \qquad\qquad\text{(E-7)}$$

$$\mathsf{checksign}(x, \mathsf{vk}(id), \mathsf{sign}(x, id)) = \mathsf{ok} \qquad\qquad\text{(E-8)}$$

$$\mathsf{checkpfk}_1(\mathsf{vk}(id), ball, \mathsf{pfk}_1(id, r, x, ball)) = \mathsf{ok} \qquad\qquad\text{(E-9)}$$
$$\text{where } ball = \mathsf{penc}(x, r, k_p)$$

$$\mathsf{checkpfk}_2(\mathsf{vk}(id), x, ball, \mathsf{pfk}_2(\mathsf{vk}(id), k, b, x, ball)) = \mathsf{ok} \qquad\qquad\text{(E-10)}$$
$$\text{where } ball = \mathsf{blind}(\mathsf{renc}(x, k), b)$$

Figure 6: Equations for encryption, blinding, signature and proofs of knowledge.

associated to a secret id I. The specific coding function used by the receipt generator for a voter with secret id I, applied to a message M is represented by $\mathsf{d}(\mathsf{p}(I), M)$. It corresponds to the function $d_V(M)$ explained in Section 1.2. The term $\mathsf{blind}(M, N)$ represents the message M blinded by N. Unblinding such a blinded term P, using the same blinding factor N is denoted by $\mathsf{unblind}(P, N)$. The term $\mathsf{penc}(M, N, P)$ refers to the encryption of plaintext M using randomness N and public key P. The term $M \circ N$ denotes the homomorphic combination of ciphertexts and the corresponding operation is written $P \diamond Q$ on plaintexts and $R * S$ on random factors. The decryption of the ciphertext C using secret key K is denoted $\mathsf{dec}(C, K)$. The term $\mathsf{renc}(M, K)$ is the re-encryption of the ciphertext M using a secret key K. The addition of secret keys is denoted by $K + L$. The term $\mathsf{sign}(M, N)$ refers to the signature of the message M using secret id N. The term $\mathsf{pfk}_1(M, N, P, Q)$ models a proof of knowledge, linked to the public identity M, that proves that Q is a ciphertext of the plaintext P using randomness N. It models what is provided to convince a prover, and its arguments represents the material needed to construct the proof itself. The term $\mathsf{pfk}_2(M, N, P, Q, R)$ represents a proof of knowledge linked to the public identity M, that R is a blinding of a re-encryption of a term Q using the secret key N and the blinding factor P. $\mathsf{pair}(M, N)$ represents the tuple (M, N). For simplicity, $\mathsf{pair}(M_1, \mathsf{pair}(\ldots, \mathsf{pair}(M_{n-1}, M_n)))$ may be abbreviated as $\langle M_1, \ldots, M_n \rangle$ and $\mathsf{fst}(\mathsf{snd}(M)^{i-1})$ as $\Pi_i(M)$ with $i \in \mathbb{N}$.

The properties of the primitives are then modeled by equipping the signature with an equational theory E that asserts that functions $+$, $*$, $\circ$ and $\diamond$ are commutative and associative, and includes the equations defined in Figure 6. The first two equations are quite standard and models left and right projections of a pair of elements. The third equation simply represents usual decryption of a ciphertext using the corresponding secret key, while Equation (E-4) reflects that a blinded ciphertext can be decrypted, yielding the corresponding blinded plaintext. Equation (E-5) models the homomorphic combination of ciphertexts. Equation (E-6) represents the re-encryption of a ciphertext. The operation of unblinding is described with Equation (E-7). Equations (E-8), (E-9), and (E-10) correspond to the verification of respectively signature and proofs of knowledge pfk_1 and pfk_2.

The rewriting system corresponding to this equational theory is AC-convergent. This can be proved showing that the system is both AC-confluent and AC-terminating. The first property is true since there is no critical pairs. AC-termination can be shown through a special measure for length of terms $|.|$ defined as follows:

$$|M| = \begin{cases} 1 & \text{if } M \text{ is a name or a variable,} \\ 2 + |M_1| + |M_2| + |M_3| & \text{if } M = \mathsf{penc}(M_1, M_2, M_3), \\ 2 + |M_1| + |M_2| & \text{if } M = \mathsf{renc}(M_1, M_2), \\ 1 + \sum_{i=1}^{k} |M_i| & \text{otherwise, i.e. } M = f(M_1, \ldots, M_k). \end{cases}$$

Using this measure, it is easy to check that the length of terms is strictly decreasing at each step of the rewriting, which ensures AC-termination of the rewriting system.

3.2 Norwegian Protocol Process Specification

We present here our model of the Norwegian voting protocol. Each player is modeled as an independent subprocess, and will be instantiated as a part of the whole protocol.

3.2.1 Voting process

The process $V(c_{bal}, c_{rec}, c_{pub}, k_{pub}, id_{vot}, pid_{rec}, v)$ represents both the voter and his computer.

$V(c_{bal}, c_{rec}, c_{pub}, k_{pub}, id_{vot}, pid_{rec}, v) = \nu\, r$.
    let $e = \mathsf{penc}(v, r, k_{pub})$, $p = \mathsf{pfk}_1(id_{vot}, r, v, e)$, $si = \mathsf{sign}(\langle e, p\rangle, id_{vot})$ in
    $\overline{c_{pub}}\langle\langle \mathsf{vk}(id_{vot}), e, p, si\rangle\rangle$.   % Public information.
    $\overline{c_{bal}}\langle\langle \mathsf{vk}(id_{vot}), e, p, si\rangle\rangle$.   % Encrypted ballot sent to B.
    $\overline{c_{rec}}\langle \mathsf{ok}\rangle$.   % Synchronization for R.
    $c_{rec}(x_r)$. $c_{bal}(x_b)$.   % Wait for inputs from R and B.
    if $\phi_{\mathsf{V}}(id_{vot}, pid_{rec}, v, e, p, si, x_b, x_r)$ then $\overline{c_{pub}}\langle \mathsf{ok}\rangle$. $\overline{c_{rec}}\langle \mathsf{ok}\rangle$

with
$$\phi_{\mathsf{V}}(id_{vot}, pid_{rec}, v, e, p, si, x_b, x_r) = (\mathsf{d}(\mathsf{p}(id_{vot}), \mathsf{blind}(v, \mathsf{s}(id_{vot}))) = x_r)$$
$$\wedge\, (\mathsf{checksign}(\mathsf{hash}(\langle \mathsf{vk}(id_{vot}), e, p, si\rangle), pid_{rec}, x_b) = \mathsf{ok}).$$

    Parameter v represents voter's vote and c_{bal}, c_{rec} denote the authenticated channels shared with, respectively, the ballot box and the receipt generator. k_{pub} represents the public key of the election used to encrypt votes; id_{vot} is the secret id of the voter and pid_{rec} is the verification key of the receipt generator. Note that messages sent over c_{bal} and c_{rec} are also sent on the public channel c_{pub}. This simulates the fact that c_{bal} and c_{rec} are authenticated but not confidential channels. The synchronization step is only here to simplify the study in the case where B is corrupted. The formula $\phi_{\mathsf{V}}(id_{vot}, pid_{rec}, v, e, p, si, x_b, x_r)$ models all the checks performed by the voters: the message received from the ballot box should be properly signed and the message from the out-of-band channel should correspond to the right receipt code.

3.2.2 Ballot box

We represent the Ballot box, ready to listen to n voters, by the process B_n defined as follows.

$B_n(c_{rec}, c_{dec}, c_{aud}, c_{pub}, k_{sec}, pid_{rec}, c_{vot}^1, pid_{vot}^1, s_{vot}^1, \ldots, c_{vot}^n, pid_{vot}^n, s_{vot}^n) =$
    $c_{vot}^1(x_1)$. BB_1 . $\ldots$. $c_{vot}^n(x_n)$. BB_n . % Processes incoming votes.
    $\overline{c_{bd}}\langle \Pi_2(x_1)\rangle$. $\ldots$. $\overline{c_{bd}}\langle \Pi_2(x_n)\rangle$. % Outputs encrypted votes to D.
    $\overline{c_{ba}}\langle x_1\rangle$. $\ldots$. $\overline{c_{ba}}\langle x_n\rangle$ % Outputs content to A.

with:

$BB_i =$
    if $\phi_{\mathsf{B}}(pid_{vot}^i, x_i)$ then   % Checks ballot's validity.
    let $b_i = \mathsf{blind}(\mathsf{renc}(\Pi_2(x_i), k_{sec}), s_{vot}^i)$ in % Computes re-encrypted blinded
    let $pok_i = \mathsf{pfk}_2(pid_{vot}^i, k_{sec}, s_{vot}^i, \Pi_2(x_i), b_i)$ in ballot and corresponding proof.
    $\overline{c_{rec}}\langle\langle x_i, b_i, pok_i\rangle\rangle$. $c_{rec}(y_i)$. % Message sent to R. Wait for R.
    if $\phi_{\mathsf{S}}(pid_{rec}, x_i, y_i)$ then $\overline{c_{pub}}\langle y_i\rangle$. $\overline{c_{vot}^i}\langle y_i\rangle$ % Checks confirmation's validity and
        sends confirmation to the Voter.

and
$$\phi_{\mathsf{S}}(M, N, U) = (\mathsf{checksign}(\mathsf{hash}(N), M, U) = \mathsf{ok})\,,$$
$$\phi_{\mathsf{B}}(V, W) = (W = \langle W_1, W_2, W_3, W_4\rangle) \wedge (\mathsf{checkpfk}_1(W_1, W_2, W_3) = \mathsf{ok})$$
$$\wedge\, (W_1 = V) \wedge (\mathsf{checksign}(\langle W_2, W_3\rangle, W_1, W_4) = \mathsf{ok})$$

where $(X = \langle X_1, \ldots, X_n \rangle)$ denotes the formula that holds only when X is a n-tuple. For example, $(X = \langle X_1, X_2 \rangle)$ denotes the formula $X = \mathsf{pair}(\mathsf{fst}(X), \mathsf{snd}(X))$.

Intuitively, we assume the ballots to be received from the authenticated channels $c_{vot}^1, \ldots, c_{vot}^n$. The Ballot box processes each ballot one after another, exchanging with the Receipt generator through the secure channel c_{rec}, before sending back a confirmation to the Voter. Once all votes have been casted, the Ballot box outputs the encrypted votes to the Decryption device using the secure channel c_{dec}, and its content to the Auditor through the secure channel c_{aud}. k_{sec} is the secret key known by B, while $pid_{vot}^1, \ldots, pid_{vot}^n$ are the public identities of the voters (*i.e.* their verification keys) and $s_{vot}^1, \ldots, s_{vot}^n$ the corresponding blinding factors.

3.2.3 Receipt generator

$R_n(c_{bal}, c_{aud}, c_{dec}, c_{pub}, k_{sec}, id_{rec}, c_{vot}^1, pid_{vot}^1, p_{vot}^1, \ldots, c_{vot}^n, pid_{vot}^n, p_{vot}^n)$ is the Receipt generator's process. It exchanges messages with the Ballot box, the Decryption service (only for an ad-hoc synchronization, used to simplify the proof) and the Auditor through secure channels c_{bal}, c_{dec}, and c_{aud} respectively. It also talks directly to voters through out-of-band channels $c_{vot}^1, \ldots, c_{vot}^n$, which are modeled as authenticated channels here. k_{sec} is the secret key received by R during the setup phase, $pid_{vot}^1, \ldots, pid_{vot}^n$ are the public identities of the voters and the corresponding receipt coding functions are $p_{vot}^1, \ldots, p_{vot}^n$.

$R_n(c_{bal}, c_{aud}, c_{dec}, c_{pub}, k_{sec}, id_{rec}, c_{vot}^1, pid_{vot}^1, p_{vot}^1, \ldots, c_{vot}^n, pid_{vot}^n, p_{vot}^n) =$
$\quad c_{vot}^1(sync_r^1) . RG_1 . c_{vot}^1(sync_v^1) . \qquad\qquad$ % Processes re-encrypted votes into receipts.
$\quad \ldots$
$\quad c_{vot}^n(sync_r^n) . RG_n . c_{vot}^n(sync_v^n) .$
$\quad \overline{c_{aud}}\langle\langle pid_{vot}^1, hbr_1 \rangle\rangle . \ldots . \overline{c_{aud}}\langle\langle pid_{vot}^n, hbr_n \rangle\rangle . \quad$ % Outputs content to A.
$\quad \overline{c_{dec}}\langle\mathsf{ok}\rangle \qquad\qquad\qquad\qquad\qquad\qquad\qquad$ % Ad-hoc synchronization for D.

with:

$RG_i =$
$\quad c_{bal}(x_i) . \qquad\qquad\qquad\qquad\qquad\qquad\quad$ % Waiting input from the Ballot box.
$\quad \text{if } \phi_{\mathsf{R}}(ipid_{vot}^i, x_i) \text{ then} \qquad\qquad\qquad\quad$ % Checks Ballot box's computations.
$\quad \text{let } r_i = \mathsf{d}(p_{vot}^i, \mathsf{dec}(\Pi_2(x_i), k_{sec})) \text{ in} \qquad$ % Computes receipt for the i-th Voter.
$\quad \text{let } hbr_i = \mathsf{hash}(\Pi_1(x_i)) \text{ in}$
$\quad \text{let } si_i = \mathsf{sign}(hbr_i, id_{rec}) \text{ in} \qquad\qquad\quad$ % Computes i-th confirmation for the Ballot box.
$\quad \overline{c_{bal}}\langle si_i \rangle . \overline{c_{pub}}\langle r_i \rangle . \overline{c_{vot}^i}\langle r_i \rangle \qquad\qquad\quad$ % Outputs to intended recipients.

and $\quad \phi_{\mathsf{R}}(X, Y) = (Y = \langle Y_1, Y_2, Y_3 \rangle) \wedge (Y_1 = \langle W_1, W_2, W_3, W_4 \rangle) \wedge (W_1 = X)$
$\qquad\qquad\quad \wedge (\mathsf{checksign}(\langle W_2, W_3 \rangle, W_1, W_4) = \mathsf{ok}) \wedge (\mathsf{checkpfk}_1(W_1, W_2, W_3) = \mathsf{ok})$
$\qquad\qquad\quad \wedge (\mathsf{checkpfk}_2(W_1, W_2, Y_2, Y_3) = \mathsf{ok}) .$

Note that instructions $c_{vot}^i(sync_r^i)$ and $c_{vot}^i(sync_v^i)$ are used to force the Receipt Generator to fully process each receipt before accepting a new entry from the Ballot box, which eases the security analysis. Note also that we slightly simplify the behavior of the Receipt Generator. [23] indicates that the Receipt Generator not only computes hbr_i as defined above but also the hash of the voter's ballot (obtained when computing $\Pi_1(x_i)$) from which the signature is dropped. We ignore the second hash as it contains less information.

3.2.4 Decryption service

The Decryption service is represented by the process $D_n(c_{bal}, c_{rec}, c_{aud}, c_{pub}, k_{sec})$. It communicates securely with the Ballot box, the Receipt generator (waiting for synchronization), and the Auditor through respectively channels c_{bal}, c_{rec}, and c_{aud}. The result is published on the public channel c_{pub}. In order to decrypt ballots, it needs to know the secret key k_{sec}. The parallelism at the end of the process models that the votes are shuffled. For simplicity, we omit the proof of correct decryption provided by the Decryption device as it should not affect privacy.

$$D_n(c_{bal}, c_{rec}, c_{aud}, c_{pub}, k_{sec}) =$$

$\quad c_{rec}(sync_d^1)$.   % Waits for R's signal to begin the tallying phase.

$\quad c_{bal}(d_1)$ . ... . $c_{bal}(d_n)$.   % Inputs encrypted votes from B.

$\quad \overline{c_{aud}}\langle \mathsf{hash}(\langle d_1, \ldots, d_n \rangle)\rangle$.   % Outputs hashed tally for A.

$\quad c_{aud}(sync_d^2)$.   % Waits A approval before processing the outcome.

$\quad (\overline{c_{pub}}\langle \mathsf{dec}(d_1, k_{sec})\rangle \mid \cdots \mid \overline{c_{pub}}\langle \mathsf{dec}(d_n, k_{sec})\rangle)$ % Publishes the results in a non-deterministic order.

3.2.5 Auditor

Finally, the Auditor is modeled by the process $A_n(c_{bal}, c_{rec}, c_{dec})$ which communicates with the other infrastructure players (Ballot box, Receipt generator, and Decryption device) using secure channels c_{bal}, c_{rec}, and c_{dec}.

$$A_n(c_{bal}, c_{rec}, c_{dec}) =$$

$\quad c_{rec}(h_1)$ . ... . $c_{rec}(h_n)$.   % Inputs from R, D and B.

$\quad c_{dec}(h_d)$. $c_{bal}(x_1)$ . ... . $c_{bal}(x_n)$.

$\quad$ if $\phi_{\mathsf{A}}(h_d, h_1, \ldots, h_n, x_1, \ldots, x_n)$ then $\overline{c_{dec}}\langle \mathsf{ok}\rangle$ else 0 % Checks and sends approval.

with:

$$\phi_{\mathsf{A}}(H, X_1, \ldots, X_n, Y_1, \ldots, Y_n) = (\mathsf{hash}(\langle \Pi_2(Y_1), \ldots, \Pi_2(Y_n)\rangle) = H)$$

$$\bigwedge_{i=1}^{n} \Big[(Y_i = \langle W_1, W_2, W_3, W_4 \rangle) \wedge (X_i = \langle Z_1, Z_2 \rangle) \wedge (W_1 = Z_1)$$

$$\wedge (\mathsf{hash}(Y_i) = Z_2) \wedge (\mathsf{checksign}(\langle W_2, W_3 \rangle, W_1, W_4) = \mathsf{ok}) \Big] .$$

3.2.6 Norwegian protocol and corruption scenarios

The interaction of all the players is simply modeled by considering all the processes in parallel, with the correct instantiation and restriction of the parameters. In what follows, the restricted names $\mathsf{a_1}$, $\mathsf{a_2}$, $\mathsf{a_3}$ model the private keys used in the protocol and the corresponding public keys $\mathsf{pk(a_1)}$, $\mathsf{pk(a_2)}$ and $\mathsf{pk(a_3)}$ are added in the process frame. The restricted names $\mathsf{c_1}$, $\mathsf{c_2}$ (resp. $\mathsf{c_{RV_1}}$ and $\mathsf{c_{RV_2}}$) model authentic channels between the two honest voters and the Ballot box (resp. the Receipt generator). The restricted names $\mathsf{id_1}$, $\mathsf{id_2}$, $\mathsf{id_R}$ represent the secret ids of honest voters and of the Receipt generator. The corresponding public id's are added to the process frame.

The process corresponding to the situation where all the authorities are honest is $P_n\,[\_]$ where n is the number of voters and the hole is the voter's place and is defined as follows:

$$P_n\,[\_] = \nu \tilde{n}.(\text{ let } \mathsf{a_3} = \mathsf{a_1} + \mathsf{a_2} \text{ in }).\Big[\_$$

$$\mid B_n(\mathsf{c_{BR}}, \mathsf{c_{BD}}, \mathsf{c_{BA}}, \mathsf{c_{out}}, \mathsf{a_2}, \mathsf{idp_R}, \mathsf{c_1}, \mathsf{idp_1}, \mathsf{s(id_1)}, \ldots, \mathsf{c_n}, \mathsf{idp_n}, \mathsf{s(id_n)})$$

$$\mid R_n(\mathsf{c_{BR}}, \mathsf{c_{RA}}, \mathsf{c_{RD}}, \mathsf{c_{out}}, \mathsf{a_3}, \mathsf{id_R}, \mathsf{c_{RV_1}}, \mathsf{idp_1}, \mathsf{p(id_1)}, \ldots, \mathsf{c_{RV_n}}, \mathsf{idp_n}, \mathsf{p(id_n)})$$

$$\mid D_n(\mathsf{c_{BD}}, \mathsf{c_{RD}}, \mathsf{c_{DA}}, \mathsf{c_{out}}, \mathsf{a_1}) \mid A_n(\mathsf{c_{BA}}, \mathsf{c_{RA}}, \mathsf{c_{DA}}) \mid \Gamma \Big]$$

with $\tilde{n} = \mathsf{a_1}, \mathsf{a_2}, \mathsf{a_3}, \mathsf{id_1}, \mathsf{id_2}, \mathsf{r_1}, \mathsf{r_2}, \mathsf{id_R}, \mathsf{c_1}, \mathsf{c_2}, \mathsf{c_{RV_1}}, \mathsf{c_{RV_2}}, \mathsf{c_{BA}}, \mathsf{c_{RA}}, \mathsf{c_{DA}}, \mathsf{c_{BR}}, \mathsf{c_{BD}}, \mathsf{c_{RD}}$ the set of restricted names and the frame $\Gamma = \{\mathsf{pk(a_i)}/\mathsf{g_i} \mid i = 1..3\} \mid \{\mathsf{vk(id_i)}/\mathsf{idp_i} \mid i = 1, 2\} \mid \{\mathsf{vk(id_R)}/\mathsf{idp_R}\}$. This frame represents the initial knowledge of the attacker: it has access to the public keys of the authorities and the verification keys of the voters. Moreover, since only the two first voters are assumed to be honest, only their two secret ids are restricted (in $\tilde{n}$). The attacker has therefore access to the secret ids of all the other voters.

The process P_n corresponds therefore to a scenario where all the election authorities are honest. To model the case where the Ballot box is corrupted, we simply provide the attacker with the Ballot box's

secrets. Formally, we define the process P_n^b defined as follows.

$$P_n^b\,[\_] = \nu\tilde{n}_b.(\text{ let } a_3 = a_1 + a_2 \text{ in }).\Big[\,\_$$
$$\mid R_n(c_{BR}, c_{RA}, c_{RD}, c_{out}, a_3, id_R, c_{RV_1}, idp_1, p(id_1), \ldots, c_{RV_n}, idp_n, p(id_n))$$
$$\mid D_n(c_{BD}, c_{RD}, c_{DA}, c_{out}, a_1) \mid A_n(c_{BA}, c_{RA}, c_{DA}) \mid \Gamma_b\Big]$$

with $\tilde{n}_b = a_1, a_3, id_1, id_2, r_1, r_2, id_R, c_{RV_1}, c_{RV_2}, c_{RA}, c_{DA}, c_{RD}$ the set of restricted names and $\Gamma_b = \{{}^{pk(a_i)}/_{g_i} \mid i = 1..3\} \mid \{{}^{vk(id_i)}/_{idp_i}, {}^{s(id_i)}/_{s_i} \mid i = 1, 2\} \mid \{{}^{vk(id_R)}/_{idp_R}\}$. Compared to P_n, we have simply removed B_n from the process (since the Ballot box is now under the control of the attacker) and the secret key a_2 and the authenticated channels $c_1, c_2, c_{BA}, c_{BR}, c_{BD}$ are now public (they are not part of the set of restricted names anymore). Finally, we have added $s(id_1), s(id_2)$ to the frame representing the initial knowledge. This models the fact that now, all the secrets of B_n are known to the attacker.

4 Formal Analysis of Ballot Secrecy

Our analysis shows that the Norwegian e-voting protocol preserves ballot secrecy, even when the Ballot box and all but two voters are corrupted, provided that the other components are honest. This of course implies ballot secrecy if all the authorities are honest and some voters are corrupted. Conversely, we identified several cases of corruption that are subject to attacks. Though not surprising, these cases were not explicitly mentioned in the literature.

In this section, we state our main security results, studying the privacy of the Norwegian protocol under two main corruptions scenarios. Moreover, we summarize existing attack scenarios. The formal proof of privacy is then detailed in the next three sections.

Ballot secrecy has been formally defined in terms of equivalence by Delaune, Kremer, and Ryan in [21]. A protocol with process $V(v, id)$ and authority process A preserves *ballot secrecy* if an attacker cannot distinguish when votes are swapped, *i.e.* it cannot distinguish when a voter a_1 votes v_1 and a_2 votes v_2 from the case where a_1 votes v_2 and a_2 votes v_1. This is formally specified by:

$$\nu\tilde{n}.\Big(A \mid V(v_1, a_1) \mid V(v_2, a_2)\Big) \approx_l \nu\tilde{n}.\Big(A \mid V(v_2, a_1) \mid V(v_1, a_2)\Big)$$

Proving ballot secrecy of the Norwegian protocol therefore amounts in proving equivalence of the corresponding processes we detailed in Section 3.

4.1 Corrupted Ballot Box and Corrupted Voters

Our main result states that the Norwegian protocol specification satisfies ballot secrecy even if the Ballot box and $n - 2$ voters are corrupted, provided that the other components are honest.

Theorem 1. *Let n be an integer representing the number of voters. Let P_n^b be the process defined in Section 3.2.6, that corresponds to the voting process where all the authorities but the Ballot box are honest. Then,*

$$P_n^b\,[V_1(a) \mid V_2(b)] \approx_l P_n^b\,[V_1(b) \mid V_2(a)]$$

with $V_i(v_j) = V(c_i, c_{RV_i}, c_{out}, g_1, id_i, idp_R, v_j)$ which corresponds to the i-th voter voting v_j.

In order to prove Theorem 1, we need to "guess" a symmetric relation $\mathcal{R}$ on closed processes satisfying the properties of a labelled bisimilarity. This amounts into describing symbolically all the possible (co-)evolutions of the two processes, depending on the actions of the adversary. The description of this relation is given in Section F. The first step of the proof consists in showing $\mathcal{R}$ satisfies property (1) of a bisimilarity relation, that is, we need to show that all pairs of frames obtained in the relation $\mathcal{R}$ are in static equivalence. In fact, all these frames are included in the final frames (modulo a "cleaning" step performed using Lemma 12 presented in appendix) corresponding to the complete execution of the two processes. It is therefore sufficient to prove static equivalence of the two final frames.

We consider the following frames:

$$\theta_{\mathsf{init}} = \{{}^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R}\} \mid \{{}^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid \mathsf{k} = 1..3\},$$

$$\theta_0 = \theta_{\mathsf{init}} \mid \{{}^{\mathsf{penc}(\mathsf{v}_k, \mathsf{r}_k, \mathsf{g}_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k, \mathsf{t}_k, \mathsf{v}_k, \mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k, \mathsf{p}_k \rangle, \mathsf{id}_k)}/_{\mathsf{si}_k} \mid \mathsf{k} = 1..2\},$$

$$\theta_k = \theta_{k-1} \mid \{{}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(M_k)), \mathsf{id}_R)}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_k), \mathsf{dec}(\Pi_2(M_k), \mathsf{a}_3))}/_{rec_k}\},$$

$$\theta_\delta = \theta_n \mid \{{}^{\mathsf{dec}(U_{\delta(k)}, \mathsf{a}_1)}/_{res_k} \mid \mathsf{k} = 1..n\}.$$

where M_i and U_k are free terms such that $\mathsf{fv}(M_{i+1}) \subseteq \mathsf{dom}(\theta_i)$ and $\mathsf{fv}(U_{k+1}) \subseteq \mathsf{dom}(\theta_n)$. Intuitively, the M_i and U_k are the recipes sent by the adversary. The restriction on the variables makes sure the adversary only use terms he has access to, at this step. δ is a substitution of $[\![1, n]\!]$ intuitively corresponding to the shuffling of the votes at the end of the election. Then each frame can be interpreted as follows:

- θ_{init} corresponds to the initial knowledge of the attacker. It contains the public data of the honest voters and the public keys of the election.

- θ_0 corresponds to the submission of ballots from the two honest voters. Note that our synchronization phase ensures that honest voters vote first. And we can show that the adversary cannot interfere with these two ballots.

- θ_k corresponds to the knowledge of the adversary once the k-th voter has voted. Intuitively, the adversary will submit any ballot he wishes (M_k) based on his prior knowledge and in return, he receives the receipt and the signature from the Receipt generator, that is, he receives $\mathsf{d}(\mathsf{p}(\mathsf{id}_k), \mathsf{dec}(\Pi_2(M_k), \mathsf{a}_3))$ and $\mathsf{sign}(\mathsf{hash}(\Pi_1(M_k)), \mathsf{id}_R)$.

- Then θ_δ corresponds to the frame with the final decryption of the votes, after shuffling, that is, the adversary can see the votes in clear after some permutation δ.

Proposition 1. *Let δ is a substitution of $[\![1, n]\!]$ and ${}^t\delta = \delta \circ [1 \mapsto 2, 2 \mapsto 1]$. Let θ_δ be the frame as defined above. Then we have:*

$$\nu\tilde{\omega}.\theta_\delta\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\theta_{{}^t\delta}\sigma_{\mathsf{R}}, \text{ with } \sigma_{\mathsf{L}} = \{{}^{\mathsf{a}}/_{\mathsf{v}_1}, {}^{\mathsf{b}}/_{\mathsf{v}_2}\} \text{ and } \sigma_{\mathsf{R}} = \{{}^{\mathsf{b}}/_{\mathsf{v}_1}, {}^{\mathsf{a}}/_{\mathsf{v}_2}\}.$$

This proposition is the main core result to establish Theorem 1.

Proof sketch. The proof is done step by step. First, we show that $\nu\tilde{\omega}.\theta_0\sigma_L \approx_s \nu\tilde{\omega}.\theta_0\sigma_R$, that is, the frames are in static equivalence once the two honest voters have voted (Lemma 13 detailed in appendix). We then show that the receipt sent by the receipt generator does not break static equivalence. This requires to prove in particular that adding the signature of a known term does preserve static equivalence (Lemma 8, one of our core lemma). Finally, we conclude by showing that the decryption of the (shuffled) vote only yields already known terms, built using the same recipes, in both frames.

The full proof of Proposition 1 can be found in Section 6. □

It then remains to show that we did not miss any possible execution, that is, show that $\mathcal{R}$ is a bisimilarity relation, which is ensured by the following proposition.

Proposition 2. *Let $\mathcal{R}$ be the relation defined in Definition 15 (Section F). Then, $\mathcal{R}$ is verifying properties* (2) *and* (3) *of Definition 4.*

Proof sketch. Given $(P, Q) \in \mathcal{R}$, we consider all possible evolutions of P in P' and show that there exists Q' such that (P', Q') remains in $\mathcal{R}$. In other words, we check that we did not forget any case when defining $\mathcal{R}$. The detailed proof is not really technical but is rather tedious and is therefore deferred to Section F. □

Theorem 1 then easily follows from Proposition 1 and Proposition 2.

4.2 Honest Authorities and Corrupted Voters

The Norwegian E-Voting Protocol specification *a fortiori* satisfies ballot secrecy even if $n - 2$ voters are corrupted, provided that the other components are honest.

Theorem 2. *Let n be an integer, that corresponds to the number of voters. Let P_n be the process defined in Section 3.2.6, that corresponds to the voting process when all authorities are honest. Then,*

$$P_n \left[V_1(\mathsf{a}) \mid V_2(\mathsf{b}) \right] \approx_l P_n \left[V_1(\mathsf{b}) \mid V_2(\mathsf{a}) \right]$$

with $V_i(\mathsf{v_j}) = V(\mathsf{c_i}, \mathsf{c_{RV_i}}, \mathsf{c_{out}}, \mathsf{g_1}, \mathsf{id_i}, \mathsf{idp_R}, \mathsf{v_j})$ which corresponds to the i-th voter voting $\mathsf{v_j}$.

Theorem 2 is a corollary of Theorem 1. While this is intuitively obvious, the use of equivalence adds some technicalities to the proof. The key proposition is that extending the initial knowledge of the attacker can only help finding attacks.

Lemma 1. *Let $\tilde{n}$ be a set of names, P, Q, two processes. Then, we have:*

$$\nu\tilde{n}. \left(P \mid \{^M/_x\} \right) \approx_l \nu\tilde{n}. \left(Q \mid \{^M/_x\} \right) \Longrightarrow \nu\tilde{n}.P \approx_l \nu\tilde{n}.Q.$$

Proof. We define a symmetric relation $\mathcal{R}$ on closed extended processes as follows:

$$\nu\tilde{n}.A \;\mathcal{R}\; \nu\tilde{n}.B \overset{\text{def}}{\Longleftrightarrow} \nu\tilde{n}. \left(A \mid \{^M/_x\} \right) \approx_l \nu\tilde{n}. \left(B \mid \{^M/_x\} \right).$$

Let us show that $\mathcal{R}$ verifies the three properties of a bisimilarity relation (*cf* Definition 4).

1. This is straightforward since $\phi(\nu\tilde{n}.A) \subseteq \phi \left(\nu\tilde{n}. \left(A \mid \{^M/_x\} \right) \right)$.

2. Let $A \rightarrow A'$. Then $\nu\tilde{n}. \left(A \mid \{^M/_x\} \right) \rightarrow \nu\tilde{n}. \left(A' \mid \{^M/_x\} \right)$. Now, since $A \;\mathcal{R}\; B$, we have that $\nu\tilde{n}. \left(A \mid \{^M/_x\} \right) \approx_l \nu\tilde{n}. \left(B \mid \{^M/_x\} \right)$ thus there exists $\overline{B}$ such that $\nu\tilde{n}. \left(B \mid \{^M/_x\} \right) \rightarrow^* \overline{B}$ and $\overline{B} \approx_l \nu\tilde{n}. \left(A' \mid \{^M/_x\} \right)$. Since B is a closed process, it can not make use of x thus it must be the case that there exists B' such that $B \rightarrow^* B'$ and $\overline{B} \equiv \nu\tilde{n}. \left(B' \mid \{^M/_x\} \right)$. Moreover since $\nu\tilde{n}. \left(A' \mid \{^M/_x\} \right) \approx_l \overline{B} \equiv \nu\tilde{n}. \left(B' \mid \{^M/_x\} \right)$, we conclude that $A' \;\mathcal{R}\; B'$.

3. Let $A \xrightarrow{\alpha} A'$. Then $\nu\tilde{n}. \left(A \mid \{^M/_x\} \right) \xrightarrow{\alpha} \nu\tilde{n}. \left(A' \mid \{^M/_x\} \right)$. Now, since $A \;\mathcal{R}\; B$, we have that $\nu\tilde{n}. \left(A \mid \{^M/_x\} \right) \approx_l \nu\tilde{n}. \left(B \mid \{^M/_x\} \right)$ thus there exists $\overline{B}$ such that $\nu\tilde{n}. \left(B \mid \{^M/_x\} \right) \rightarrow^* \xrightarrow{\alpha} \rightarrow^* \overline{B}$ and $\overline{B} \approx_l \nu\tilde{n}. \left(A' \mid \{^M/_x\} \right)$. Since A is a closed process, it does not contain x, thus we must have that the label α does not contain x either. Thus, if $\nu\tilde{n}. \left(B \mid \{^M/_x\} \right) \rightarrow^* \xrightarrow{\alpha} \rightarrow^* \overline{B}$, we must have that there exists B' such that $B \rightarrow^* \xrightarrow{\alpha} \rightarrow^* B'$ and $\overline{B} \equiv \nu\tilde{n}. \left(B' \mid \{^M/_x\} \right)$. Moreover since $\nu\tilde{n}. \left(A' \mid \{^M/_x\} \right) \approx_l \overline{B} \equiv \nu\tilde{n}. \left(B' \mid \{^M/_x\} \right)$, we conclude that $A' \;\mathcal{R}\; B'$.

Now, since $\approx_l$ is the largest relation satisfying these three properties, we conclude. $\qquad\square$

We are now ready to prove Theorem 2.

Proof of Theorem 2. Let us define $A = P_n^b \left[V_1(\mathsf{a}) \mid V_2(\mathsf{b}) \right]$ and $B = P_n^b \left[V_1(\mathsf{b}) \mid V_2(\mathsf{a}) \right]$. According to Theorem 1, we have that: $A \approx_l B$. Thus, for all closing evaluation context $C[\_]$, we also have: $C[A] \approx_l C[B]$. Let us consider the following closing evaluation context:

$$C[\_] = \nu\tilde{m}. \left[\_ \mid B_n(\mathsf{c_{BR}}, \mathsf{c_{BD}}, \mathsf{c_{BA}}, \mathsf{c_{out}}, \mathsf{a_2}, \mathsf{idp_R}, \mathsf{c_1}, \mathsf{idp_1}, \mathsf{s_1}, \dots, \mathsf{c_n}, \mathsf{idp_n}, \mathsf{s_n}) \right],$$

with $\tilde{m} = \mathsf{a_2}, \mathsf{c_1}, \mathsf{c_2}, \mathsf{c_{BR}}, \mathsf{c_{BD}}$ and B_n is defined as the honest one except that we replace arguments $\mathsf{s(id_1)}$ and $\mathsf{s(id_2)}$ by variables $\mathsf{s_1}$ and $\mathsf{s_2}$. Then:

$$C[A] = \nu\tilde{m}. \left[\nu\tilde{n}_b.(\text{ let } \mathsf{a_3} = \mathsf{a_1} + \mathsf{a_2} \text{ in }). \left(V_1(\mathsf{a}) \mid V_2(\mathsf{b}) \mid R_n \mid D_n \mid \Gamma_b \right) \sigma_L \mid B_n \right]$$

18

where B_n, R_n and D_n are just short notations of the different processes with their attributes. Since B_n is such that $(\mathsf{fv}(B_n) \cup \mathsf{fn}(B_n)) \cap \tilde{n}_b = \emptyset$, we have that:

$$C[A] = \nu\tilde{m}.\,[\nu\tilde{n}_b.(\text{ let } \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2 \text{ in }).\,(V_1(\mathsf{a}) \mid V_2(\mathsf{b}) \mid B_n \mid R_n \mid D_n \mid \Gamma_b)]$$
$$= \nu(\tilde{m}, \tilde{n}_b).(\text{ let } \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2 \text{ in }).\,[V_1(\mathsf{a}) \mid V_2(\mathsf{b}) \mid B_n \mid R_n \mid D_n \mid \Gamma_b]\,.$$

But $(\tilde{m}, \tilde{n}_b) = \tilde{n}$ and:

$$B_n(\mathsf{c}_{\mathsf{BR}}, \mathsf{c}_{\mathsf{BD}}, \mathsf{c}_{\mathsf{BA}}, \mathsf{c}_{\mathsf{out}}, \mathsf{a}_2, \mathsf{idp}_{\mathsf{R}}, \mathsf{c}_1, \mathsf{idp}_1, \mathsf{s}_1, \ldots, \mathsf{c}_n, \mathsf{idp}_n, \mathsf{s}_n)\Gamma_b$$
$$= B_n(\mathsf{c}_{\mathsf{BR}}, \mathsf{c}_{\mathsf{BD}}, \mathsf{c}_{\mathsf{BA}}, \mathsf{c}_{\mathsf{out}}, \mathsf{a}_2, \mathsf{idp}_{\mathsf{R}}, \mathsf{c}_1, \mathsf{idp}_1, \mathsf{s}(\mathsf{id}_1), \ldots, \mathsf{c}_n, \mathsf{idp}_n, \mathsf{s}(\mathsf{id}_n))\Gamma_b\,.$$

So, we get that $C[A] = \overline{P}_n[V_1(\mathsf{a}) \mid V_2(\mathsf{b})]$, where $\overline{P}_n^a$ is exactly the same as P_n except that Γ is replaced by Γ_b. Doing the same for $C[B]$, we deduce that $C[B] = \overline{P}_n[V_1(\mathsf{b}) \mid V_2(\mathsf{a})]$ and thus $\overline{P}_n[V_1(\mathsf{a}) \mid V_2(\mathsf{b})] \approx_l \overline{P}_n[V_1(\mathsf{b}) \mid V_2(\mathsf{a})]$. According to Lemma 1, this equivalence implies that:

$$P_n[V_1(\mathsf{a}) \mid V_2(\mathsf{b})] \approx_l P_n[V_1(\mathsf{b}) \mid V_2(\mathsf{a})]\,.$$

$\square$

4.3   Attacks

We have shown that the Norwegian protocol guarantees ballot privacy provided that the Receipt generator, the Decryption service, and the Auditor are honest. We review further cases of corruption where ballot secrecy is no longer guaranteed.

Dishonest Decryption service.   The Decryption service is a very sensitive component since it has access to the decryption key a_1 of the public key used for the election. Therefore, a corrupted Decryption service can very easily decrypt all encrypted ballots and thus learns the votes as soon as he has access to the communication between the voters and the Ballot box. Such an attack is not possible in [23, 24] since communications are assumed to be secure (e.g. using TLS channels). It is interesting to note that the combination of both studies indicates that in case the Decryption service is corrupted then ballot secrecy solely relies on the security of the communication channels between the voters and the Ballot box.

Dishonest Ballot box and Receipt generator.   Clearly, if the Ballot box and the Receipt generator collude, they can compute $\mathsf{a}_1 = \mathsf{a}_3 - \mathsf{a}_2$ and they can then decrypt all incoming encrypted ballots. More interestingly, a corrupted Receipt generator does not need the full cooperation of the ballot box for breaking ballot secrecy. Indeed, assume that the Receipt generator has access, for some voter V, to the blinding factor s_V used by the Ballot box to blind the ballot. Recall that the Receipt generator retrieves $f(o)^{s_V}$ when generating the receipt codes (by computing $\check{w}\check{x}^{-\mathsf{a}_3}$). Therefore, the Receipt generator can compute $\mathsf{f}(\mathsf{o}')^{s_V}$ for any possible voting option o'. Comparing with the obtained values with $\mathsf{f}(\mathsf{o})^{s_V}$ it would easily deduce the chosen option o. Of course, the more blinding factors the Receipt generator can get, the more voters it can attack. Therefore, the security of the protocol strongly relies on the security of the blinding factors which generation and distribution are left unspecified in the documentation (who has access to the data during the generation? How the data are distributed and how secure the distribution is?). The Ballot box can also perform a similar attack provided that it has access to the SMS sent by the Receipt generator and provided it can learn the coding function d_V for some voters V (which probably requires partial corruption of the Receipt generator as well). Note that if the Ballot box and the Receipt generator collude, verifiability is broken as well, as pointed in [23, 24], since votes can be changed without voters noticing.

Dishonest Ballot box and Auditor.   Even if the Auditor does not hold any secret (besides the access to the output of both the Ballot box and the Receipt generator), it is still a key component in the voting process. Indeed, it ensures that the ballots sent to the Decryption service indeed correspond to the ballots sent by the voters (unless both the Ballot box and the Receipt generator are corrupted). Assume now that the Ballot box and the Auditor corrupted. Then the Ballot box can send any ballots it wants to the Decryption service

(the - corrupted - Auditor would not complain). This is a clear breach of security in terms of the correctness of the result: indeed, the results would not correspond to the votes as casted by the voters (as also pointed in [23, 24]). As a consequence, this is also a breach of ballot privacy. Indeed, the Ballot box may send the same ballot several times (possibly re-randomized) therefore obtaining a bias of information about the vote casted by the voter under attack.

5 Lemmas for Static Equivalence

The proof of our main result requires several intermediate lemmas. We believe that some of them are of independent of interest. We first state lemmas that are independent of the equational theory (Section 5.1) and then one of them that relies on it but is still quite general and could be reused for other formal studies of protocols (Section 5.2).

5.1 Generic Lemmas

We use some arguments repetitively on our proofs and we found useful to state them separately. First, we notice that splitting pairs preserves static equivalence.

Lemma 2. *Let $\tilde{n}, \tilde{m}$ be names, θ_1, θ_2 substitutions and let us define $\phi = \nu\tilde{n}.\left(\theta_1 \mid \{^{\langle U_1, U_2 \rangle}/_x\}\right)$ and $\psi = \nu\tilde{m}.\left(\theta_2 \mid \{^{\langle V_1, V_2 \rangle}/_x\}\right)$ two frames with U_1, U_2, V_1, V_2 some terms. We have:*

$$\phi \approx_s \psi \iff \nu\tilde{n}.\left(\theta_1 \mid \{^{U_1}/_{x_1},^{U_2}/_{x_2}\}\right) \approx_s \nu\tilde{m}.\left(\theta_2 \cup \{^{V_1}/_{x_1},^{V_2}/_{x_2}\}\right).$$

Proof. Let $\phi' = \nu\tilde{n}.\left(\theta_1 \mid \{^{U_1}/_{x_1},^{U_2}/_{x_2}\}\right)$ and $\psi' = \nu\tilde{m}.\left(\theta_2 \cup \{^{V_1}/_{x_1},^{V_2}/_{x_2}\}\right)$.

Let M, N be terms s.t. $(M =_E N)\phi'$. We consider $\delta = [x_1 \mapsto \Pi_1(x), x_2 \mapsto \Pi_2(x)]$ and we have, for all term P: $(P\delta)\phi =_E P\phi'$ and $(P\delta)\psi =_E P\psi'$. Thus $M\phi' =_E N\phi'$ implies $(M\delta)\phi =_E (N\delta)\phi$. Since $\phi \approx_s \psi$ we deduce $(M\delta)\psi =_E (N\delta)\psi$, that is $M\psi' =_E N\psi'$. Since ϕ and ψ play a symmetric role, we deduce that $M\psi' =_E N\psi'$ implies $M\phi' =_E N\phi'$ as well. $\qquad\square$

Adding a deducible term to the frames preserves static equivalence, provided the condition that the same recipe is used in the two frames.

Lemma 3. *Let $\tilde{n}, \tilde{m}$ be names, θ_1, θ_2 substitutions, and $\phi = \nu\tilde{n}.\theta_1$ and $\psi = \nu\tilde{m}.\theta_2$ frames. Let U be a free term, we have:*

$$\phi \approx_s \psi \iff \nu\tilde{n}.\left(\theta_1 \mid \{^{U\theta_1}/_x\}\right) \approx_s \nu\tilde{m}.\left(\theta_2 \cup \{^{U\theta_2}/_x\}\right).$$

Proof. Let $\phi' = \nu\tilde{n}.\left(\theta_1 \mid \{^{U\theta_1}/_x\}\right)$ and $\psi' = \nu\tilde{m}.\left(\theta_2 \cup \{^{U\theta_2}/_x\}\right)$.

$\impliedby|$ Straightforward.

$\implies|$ Let M, N be terms such that $(M =_E N)\psi'$. We consider $\delta : x \mapsto U$ and we have, for all term P: $\overline{(P\delta)}\phi =_E P\phi'$ and $(P\delta)\psi =_E P\psi'$. Thus $M\phi' =_E N\phi'$ implies $(M\delta)\phi =_E (N\delta)\phi$. Since $\phi \approx_s \psi$, we deduce $(M\delta)\psi =_E (N\delta)\psi$, that is $M\psi' =_E N\psi'$. $\qquad\square$

The next lemmas hold for equational theories for which *destructors* can be identified.

Definition 5. *Let E be an equational theory induced by an AC convergent rewriting system $\mathcal{R}$ and Σ a signature. We say that $f \in \Sigma$ is a **destructor in** E if for any rewrite rule $l \to r$ of $\mathcal{R}$*

$$f \notin r \text{ and } (f \notin l \text{ or } l = f(l_1, \dots, l_n) \text{ and } \forall\, i \in [\![1, n]\!],\ f \notin l_i),$$

*i.e. f can only appear in head in l and does not appear in r. Terms of the form $f(t_1, \dots, t_n)$ with f destructor in E are called **destructor terms**.*

The next lemma states that destructor terms cannot be introduced by rewriting rules.

Lemma 4. *Let E be an equational theory induced by an AC-convergent rewriting system $\mathcal{R}$, Σ a signature and $f \in \Sigma$ a destructor in E. Let $C[\_]$ be a context such that $f \notin C$ and let $T_1,\ldots, T_n$ be terms in normal form. Then there exists a context $C'[\_]$ such that $f \notin C'$ and $C[T_1,\ldots,T_n] \to C'[M_1,\ldots,M_p]$ where $M_1,\ldots,M_p$ are subterms of $T_1, \ldots, T_n$.*

Proof. Let us consider such context $C[\_]$ and such terms $T_1,\ldots,T_n$. For this proof, we also consider that we flatten AC symbols, i.e. a term $\oplus(x,\oplus(y,z))$ will be considered as $\oplus(x,y,z)$ for any AC symbol $\oplus$. Moreover, we can consider w.l.o.g. that each T_i does not start with an AC-symbol. If $T_i = \oplus(N_1,\ldots,N_q)$ for one $i \in [\![1,n]\!]$, then we can consider $\overline{C}[\_]$ a new context deduced from $C[\_]$ by adding a $\oplus$ symbol at each position T_i should be inserted in $C[\_]$. Then, we have:

$$C[T_1,\ldots,T_n] =_{AC} \overline{C}[T_1,\ldots,T_{i-1},N_1,\ldots,N_q,T_{i+1},\ldots,T_n].$$

Now, if $C[T_1,\ldots,T_n]$ is in normal form, the result is straightforward. If it is not, then we know that there exists a position s, a rule $l \to r$ from $\mathcal{R}$ and a substitution θ such that $C[T_1,\ldots,T_n]|_s =_{AC} l\theta$. Let x be a variable of l and p_x be a position such that $l|_{p_x} = x$. We know that p_x can't be the root, otherwise $l = x$ and $\mathcal{R}$ would contain a rule $x \to r$. Such a rule would allow infinite chains of reduction which is in contradiction with hypothesis on $\mathcal{R}$. Thus, we can move one step up and call p_s the position such that $p_x = p_s * j$ with $j \in \mathbb{N}^*$ and $l|_{p_s} = g \in \Sigma$. We consider two cases:

- g is not an AC-symbol. We consider two subcases:

    - p_x is a position of $C[\_]$. Then $x\theta =_{AC} C_x[T_1,\ldots,T_n]$ with $C_x[\_]$ a sub context of $C[\_]$ and we can replace the substitution of x in θ by $\{^{C_x[T_1,\ldots,T_n]}/_x\}$.

    - p_x is not a position of $C[\_]$, i.e. there exists p_i and p_2 two positions such that $p_x = p_i * p_2$ with $C[T_1,\ldots,T_n]|_{p_i} =_{AC} T_i$ and $T_i|_{p_2} =_{AC} x\theta =_{AC} W$ with $W \in \mathsf{St}(T_i)$. In that case, we can just replace the substitution of x in θ by $\{^W/_x\}$.

- $g = \oplus$, an AC-symbol. Then, we have $l|_{p_s} =_{AC} x \oplus l'$. We consider again two subcases:

    - Either p_s is a position of $C[\_]$. Then, $l|_{p_s}\theta =_{AC} (x \oplus l')\theta =_{AC} C_x[T_1,\ldots,T_n]$. Thus, $\mathrm{head}(C_x[T_1,\ldots,T_n]) = \oplus$ and we can expand it as follows:

    $$C_x[T_1,\ldots,T_n] =_{AC} \overset{m_1}{\underset{i=1}{\oplus}} C_x^i[T_1,\ldots,T_n] \overset{m_2}{\underset{j=1}{\oplus}} W_j$$

    with $C_x^i[\_]$ subcontexts of $C_x[\_]$ and W_j being one of $T_1,\ldots,T_n$. Then, each $x\theta$ is linked to a subset of this sum and we can replace the substitution of x in θ by $\{^{\overset{w_1}{\underset{i=1}{\oplus}} C_x^i[T_1,\ldots,T_n] \overset{w_2}{\underset{j=1}{\oplus}} W_j}/_x\}$ with $w_1 \leq m_1$ and $w_2 \leq m_2$.

    - Or $l|_{p_s}\theta =_{AC} W$ with $W \in \mathsf{St}(T_i)$ for one $i \in [\![1,n]\!]$. Then, $x\theta$ is subterm of W and we conclude.

Finally, for each case, we have that $l\theta =_{AC} l\theta'$ where θ' contain only substitutions of variables of l by sub contexts of $C[\_]$ and subterms of $T_1,\ldots,T_n$. Thus, we have:

$$C[T_1,\ldots,T_n]|_s =_{AC} l\theta \to r\theta' =_{AC} C_r[R_1,\ldots,R_{q_r}]$$

where $C_r[\_]$ contains r and all subcontexts $C_x[\_]$, $C_x^k[\_]$. Since $f \notin C$ and $C_x[\_]$, $C_x^k[\_]$ are subcontexts of $C[\_]$, we know that $f \notin C_x$ and $f \notin C_x^k$. Moreover, since f is a destructor in E, $f \notin r$. Thus, $f \notin C_r$. Finally, $R_1,\ldots,R_{q_r}$ are subterms of $T_1,\ldots,T_n$, then:

$$C[T_1,\ldots,T_n] \to C[T_1,\ldots,T_n]\Big[C_r[R_1,\ldots,R_{q_r}]\Big]_p = C'[M_1,\ldots,M_q].$$

$\square$

If a destructor term t is non deducible and does not appear as subterm of a frame then t cannot appear as subterm of any deducible term.

Lemma 5. *Let E be an equational theory induced by an AC convergent rewriting system $\mathcal{R}$, Σ a signature and $f \in \Sigma$ is a destructor in E. Let φ be a frame and $U = f(U_1, \ldots, U_n)$ a term in normal form such that $U \notin \mathsf{St}(\varphi)$ and $\varphi \not\vdash U$. Then:*

$$\text{for all term } T \text{ in normal form such that } \varphi \vdash T, \ U \notin \mathsf{St}(T).$$

Proof. Let us prove this by induction on the number of steps used to deduce T.

Base case: $T \in \varphi$. Then, since $U \notin \mathsf{St}(\varphi)$, we have that $U \notin \mathsf{St}(T)$.

Induction case: Now suppose that for any T in normal form such that $\varphi \vdash T$ in n steps, then $U \notin \mathsf{St}(T)$. Let T in normal form such that $\varphi \vdash T$ in $n + 1$ steps. We have $\varphi \vdash T =_{AC} g(T_1, \ldots, T_n)\!\downarrow$ where $g \in \Sigma$ and $\varphi \vdash T_i$ in n steps with T_i in normal form for $i \in [\![1, n]\!]$.

- If $g(T_1, \ldots, T_n)$ is already in normal form, then we have two subcases:

  - If $g \neq f$, then according to the induction hypothesis and since $U \notin \mathsf{St}(T_i)$, we have $U \notin \mathsf{St}(T)$.

  - If $g = f$, then, if $U \in \mathsf{St}(T)$ and since $U \notin \mathsf{St}(T_i)$ according to the induction hypothesis, we must have $T = U$, which yields to a contradiction since $\varphi \vdash T$ and $\varphi \not\vdash U$.

- If $g(T_1, \ldots, T_n)$ is not in normal form. If $g = f$ then, we have that $f(T_1, \ldots, T_n) \to r[\widetilde{T_i}]$ for some rule $l \to r$. Since f is a destructor in E, we know that $f \notin r$. Thus, according to Lemma 4, we know that $r[\widetilde{T_i}] \to^* C[\widetilde{T_i}]$ such that $f \notin C$. In that case, if $U \in \mathsf{St}(C[\widetilde{T_i}])$, then, we must have $U \in \mathsf{St}(\widetilde{T_i})$ which is a contradiction. If $g \neq f$, then, using Lemma 4 with $C[T_1, \ldots, T_n] =_{AC} g(T_1, \ldots, T_n)$, we have that $\exists\, C'$ such that $f \notin C'$ and $g(T_1, \ldots, T_n) \to^* C'[\widetilde{T_i}]$ with $\widetilde{T_i}$ subterms of $T_1, \ldots, T_n$. Then $T =_{AC} C'[\widetilde{T_i}]$ but we know that $U \notin \mathsf{St}(\widetilde{T_i})$ according to induction hypothesis, thus $U \notin \mathsf{St}(T)$.

This concludes the induction. $\qquad\square$

A context in normal form applied to a destructor term remains in normal form.

Lemma 6. *Let E be an equational theory induced by an AC convergent rewriting system $\mathcal{R}$, Σ a signature and $f \in \Sigma$ is a destructor in E. Let $U = f(U_1, \ldots, U_n)$ a term in normal form and $P[X]_p$ another term in normal form. Then $P[U]_p$ is also in normal form.*

Proof. Assume that $P[U]$ is not in normal form. Since $P[X]$ and U are in normal form, it implies that $\exists\, p$ a position of $P[\_]$ such that $P[U]|_p =_{AC} l\theta$ with some rule $l \to r$ and some substitution θ. But f is a destructor in E, thus, f must appear at the head of l, which is a contradiction with the position p. $\qquad\square$

As a consequence of the previous lemmas, we can state that adding non deducible destructor terms to the frames preserve static equivalence.

Lemma 7. *Let E be an equational theory, Σ a signature and $f, g \in \Sigma$ destructors in E. Let $\varphi_1 = \nu\tilde{\omega}_1.\theta_1$ and $\varphi_2 = \nu\tilde{\omega}_2.\theta_2$ be frames. Let $U_1 = f(U_1^1, \ldots, U_p^1)$ and $U_2 = g(U_1^2, \ldots, U_q^2)$ be terms in normal form such that, $\forall\, i \in \{1, 2\}$, $\varphi_i \not\vdash U_i$ and $U_i \notin \mathsf{St}(\varphi_i)$. Then:*

$$\varphi_1 \approx_s \varphi_2 \Longleftrightarrow \nu\tilde{\omega}_1.\left(\theta_1 \mid \{^{U_1}/_x\}\right) \approx_s \nu\tilde{\omega}_2.\left(\theta_2 \mid \{^{U_2}/_x\}\right).$$

Proof. Let $\varphi_i' = \nu\tilde{\omega}_i.\left(\theta_i \mid \{^{U_i}/_x\}\right) = \nu\tilde{\omega}_i.\theta_i'$ for $i \in \{1, 2\}$.

$\Longleftarrow\!\mid$ Straightforward.

$\Longrightarrow\!\mid$ Let M and N be terms s.t. $\mathsf{fv}(M, N) \subseteq \mathsf{dom}(\varphi_1')$ and $\mathsf{bn}(M, N) \cap \mathsf{bn}(\varphi_1') = \emptyset$ with $(M =_E N)\varphi_1'$. Then:

$$
\begin{array}{rcl}
M\varphi_1' & =_E & N\varphi_1' \\
(M\varphi_1')\!\downarrow & =_{AC} & (N\varphi_1')\!\downarrow \\
(M\varphi_1\delta_1)\!\downarrow & =_{AC} & (N\varphi_1\delta)\!\downarrow
\end{array}
$$

with $\delta_1 : x \mapsto U_1$, according to φ_1' definition. Since U_1 is in normal form, we apply Lemma 6:

$$(M\varphi_1)\!\downarrow \delta_1 \quad =_{AC} \quad (N\varphi_1)\!\downarrow \delta_1$$
$$((M\varphi_1)\!\downarrow \delta_1)\delta_1^{-1} \quad =_{AC} \quad ((N\varphi_1)\!\downarrow \delta_1)\delta_1^{-1}$$

with $\delta_1^{-1} : U_1 \mapsto x$. We know that $U_1 = f(U_1^1, \ldots, U_p^1)$ with f destructor in E and U_1 in normal form and not subterm of φ_1. Thus, using Lemma 5, for any T in normal form s.t. $\varphi_1 \vdash T$, then $U_1 \notin \mathsf{St}(T)$. And, since $\varphi_1 \vdash (M\varphi_1)\!\downarrow$ (resp. $(N\varphi_1)\!\downarrow$) we have that $(M\varphi_1)\!\downarrow \delta_1^{-1} = (M\varphi_1)\!\downarrow$ (resp. $(N\varphi_1)\!\downarrow \delta_1^{-1} = (N\varphi_1)\!\downarrow$). Thus:

$$(M\varphi_1)\!\downarrow \delta_1\delta_1^{-1} \quad =_{AC} \quad (N\varphi_1)\!\downarrow \delta_1\delta_1^{-1}$$
$$(M\varphi_1)\!\downarrow \quad =_{AC} \quad (N\varphi_1)\!\downarrow$$

Since $\varphi_1 \approx_s \varphi_2$, we have (with $\delta_2 : x \mapsto U_2$):

$$(M\varphi_2)\!\downarrow \quad =_{AC} \quad (N\varphi_2)\!\downarrow$$
$$(M\varphi_2)\!\downarrow \delta_2 \quad =_{AC} \quad (N\varphi_2)\!\downarrow \delta_2$$

Using Lemma 6, we deduce $(M\varphi_2)\!\downarrow \delta_2 =_{AC} (N\varphi_2)\!\downarrow$, implying that $(M =_E N)\varphi_1' \to (M =_E N)\varphi_2'$. Repeating the same reasoning, we can also prove that $(M =_E N)\varphi_2' \to (M =_E N)\varphi_1'$, and we conclude. $\qquad\square$

5.2 A More Specific Lemma

We show that adding the signature of a deducible term preserves static equivalence, when the same recipe is used in both frames. We believe that this lemma holds for other primitives provided the equations are similar to those for the "sign" symbol like the case of zero-knowledge proofs.

Lemma 8. *Let $\varphi_1 = \nu\omega_1.\theta_1$ and $\varphi_2 = \nu\omega_2.\theta_2$ be two frames, x a fresh variable and a, a name such that $a \in \omega_1 \cap \omega_2$, $\{\mathsf{vk}(a)/_{\mathsf{idp_a}}\} \in \theta_1 \cap \theta_2$ and $\varphi_1, \varphi_2 \nvdash a$. Let $U = \mathsf{sign}(U', a)$ in normal form with U' a free term and such that $(U\varphi_i)\!\downarrow \notin \mathsf{St}(\varphi_i)$. Then:*

$$\varphi_1 \approx_s \varphi_2 \Longleftrightarrow \nu\omega_1.\left(\theta_1 \mid \{^{(U\varphi_1)\!\downarrow}/_x\}\right) \approx_s \nu\omega_2.\left(\theta_2 \mid \{^{(U\varphi_2)\!\downarrow}/_x\}\right).$$

Before proving this lemma, we introduce the notion of down-to-top strategy, where a rewrite rule is applied first as deep as possible. Note that since we consider convergent rewrite systems, we may chose any rewrite strategy.

Definition 6. *Let us consider $M_0 \xrightarrow{l_1 \to r_1} M_1 \longrightarrow \ldots \xrightarrow{l_i \to r_i} M_i \ldots \longrightarrow M_n = M_0\!\downarrow$ a reduction strategy. This strategy is called* down-to-top, *noted $\longrightarrow_{dt}$, if, for all $i \in [\![1, n]\!]$, $M_{i-1}|_{p_i} =_{AC} l_i\theta_i \to r_i\theta_i$ and for all position $q < p_i$, $M_{i-1}|_q$ is in normal form.*

Proof. Let $\overline{\varphi}_i = \nu\omega_1.\overline{\theta}_i$ with $\overline{\theta}_i = \theta_i \mid \{^{(U\varphi_i)\!\downarrow}/_x\}$ for $i \in \{1, 2\}$.

$\underline{\Longleftarrow|}$ Straightforward.

$\underline{\Longrightarrow|}$ We first assume the two following claims that we prove next.

Claim 1. *Let M be a term s.t. $\mathsf{fn}(M) \cap \omega_i = \emptyset$. Then, for any term T s.t. $M\varphi_i \longrightarrow_{dt}^* T$, we have $(U\varphi_i)\!\downarrow \notin \mathsf{St}(T)$.*

Claim 2. *Let M be a term s.t. $(U\varphi_i)\!\downarrow \notin \mathsf{St}(M)$ and for all term T s.t. $M \longrightarrow_{dt}^* T$, then $(U\varphi_i)\!\downarrow \notin \mathsf{St}(T)$. Let $\sigma = \{^{(U\varphi_i)\!\downarrow}/_x\}$ a substitution. If $M\sigma \longrightarrow_{dt}^* T$ with no rule (8) involved in the reduction path, then there exists some term T' such that $M \longrightarrow_{dt}^* T'$ and $T'\sigma =_{AC} T$.*

We now suppose that $\varphi_1 \approx_s \varphi_2$ and we consider M and N two terms s.t. $\mathsf{fv}(M, N) \subseteq \mathsf{dom}(\overline{\varphi}_1)$ and $\mathsf{fn}(M, N) \cap \omega_1 = \emptyset$. (Since $\varphi_1 \approx_s \varphi_2$, we have $\omega_1 = \omega_2$ and $\mathsf{dom}(\overline{\varphi}_1) = \mathsf{dom}(\overline{\varphi}_2)$.) We are going to prove that $(M =_E N)\overline{\varphi}_1$ iff $(M =_E N)\overline{\varphi}_2$ by induction on the number of positions p in M and N s.t. $M|_p = \mathsf{checksign}(M_1, M_2, M_3)$ and $(M|_p\overline{\varphi}_i)\!\downarrow =_E \mathsf{ok}$ for $i = 1$ or $i = 2$. We also assume down-to-top reduction strategies only (noted $\longrightarrow_{dt}$).

Base case: There is no such position in M and N. Then, we can apply Claim 2 on $M\overline{\varphi}_i$ and $N\overline{\varphi}_i$. Indeed, let $\sigma_i = \{^{(U\varphi_i)\downarrow}/_x\}$. For $P \in \{M, N\}$, then $P\overline{\varphi}_i = (P\varphi_i)\sigma_i \longrightarrow_{dt}^* (P\overline{\varphi}_i)\downarrow$. If there exists a position p such that $(P\varphi_i)|_p =_{AC} (U\varphi_i)\downarrow$ then, according to the hypothesis on φ_i, p must be a position of P (and not at a leaf). This implies that $P|_{p*2} = a$ which would be a contradiction with the fact that a is restricted. Thus, we have $(U\varphi_i)\downarrow \notin \mathsf{St}(P\varphi_i)$. We also have $P\varphi_i \longrightarrow_{dt}^* (P\varphi_i)\downarrow$ and s.t. $\mathsf{fn}(P) \cap \omega_i = \emptyset$. Using Claim 1, we deduce that $(U\varphi_i)\downarrow$ does not occur in any of the terms in the reduction $P\varphi_i \longrightarrow_{dt}^* (P\varphi_i)\downarrow$. Applying Claim 2 leads to $(P\overline{\varphi}_i)\downarrow =_{AC} (P\varphi_i)\downarrow \sigma_i$. Then:

$$(M\overline{\varphi}_i)\downarrow =_{AC} (N\overline{\varphi}_i)\downarrow \qquad\qquad (M\varphi_2)\downarrow =_{AC} (N\varphi_2)\downarrow \quad (\varphi_1 \approx_s \varphi_2)$$
$$(M\varphi_1)\downarrow \sigma_1 =_{AC} (N\varphi_1)\downarrow \sigma_1 \ \ (\text{Claim 2}) \qquad\qquad (M\varphi_2)\downarrow \sigma_2 =_{AC} (N\varphi_2)\downarrow \sigma_2$$
$$(M\varphi_1)\downarrow =_{AC} (N\varphi_1)\downarrow \qquad (\text{Claim 1}) \qquad\qquad (M\overline{\varphi}_2)\downarrow =_{AC} (N\overline{\varphi}_2)\downarrow . \qquad (\text{Claim 2})$$

Induction step: We suppose that there exists at least one position p in M or N s.t. $M|_p = \mathsf{checksign}(M_1, M_2, M_3)$ (or $N|_p$) and $(M|_p\overline{\varphi}_i)\downarrow =_E \mathsf{ok}$ (resp. $N|_p$) for $i = 1$ or $i = 2$. Let us consider w.l.o.g. that this position is in M and that it reduces when $\overline{\varphi}_1$ is applied. (We note that if $(M|_p\overline{\varphi}_2)\downarrow =_E \mathsf{ok}$ too then we have $(M[\mathsf{ok}]_p =_E M)\overline{\varphi}_i$ for $i \in \{1, 2\}$ and we can conclude using the induction hypothesis on $M[\mathsf{ok}]_p$ and N.) We consider the deepest position satisfying this in M. Thus, we know that M_1, M_2 and M_3 do not contain such a position.

Since $(M|_p\overline{\varphi}_1)\downarrow =_E \mathsf{ok}$, then, according to E, we have that $(M_3\overline{\varphi}_1)\downarrow =_{AC} \mathsf{sign}((M_1\overline{\varphi}_1)\downarrow, Q)$ and $(M_2\overline{\varphi}_1)\downarrow =_{AC} \mathsf{vk}(Q)$ for some term Q. According to the hypothesis on position p, we can apply the same reasoning as used in the base case on M_1, M_2 and M_3 separately and therefore we can apply Claim 2 on them, which leads to $(M_i\overline{\varphi}_1)\downarrow =_{AC} (M_i\varphi_1)\downarrow \sigma_1$ for $i \in \{1, 2, 3\}$. We have then the following equalities:

- $(M_1\varphi_1)\downarrow =_{AC} T_1$ and $T_1\sigma_1 =_{AC} (M_1\overline{\varphi}_1)\downarrow$.

- $(M_2\varphi_1)\downarrow =_{AC} T_2$ and $T_2\sigma_1 =_{AC} \mathsf{vk}(Q)$. According to definition of σ_1, which does not contain a term of this form, we must have $T_2 =_{AC} \mathsf{vk}(T_2')$ and $T_2'\sigma_2 =_{AC} Q$.

- $(M_3\varphi_1)\downarrow =_{AC} T_3$ and $T_3\sigma_1 =_{AC} \mathsf{sign}((M_1\overline{\varphi}_1)\downarrow, Q)$. Using the two previous equalities, we have $T_3\sigma_1 =_{AC} \mathsf{sign}(T_1\sigma_1, T_2'\sigma_1)$. Then, according to the definition of σ_1, we consider two subcases:

  – $T_3 =_{AC} \mathsf{sign}(T_1, T_2',)$ and then, we have:

  $$(M|_p\varphi_1)\downarrow =_E \mathsf{checksign}((M_1\varphi_1)\downarrow, (M_2\varphi_1)\downarrow, (M_3\varphi_1)\downarrow)\downarrow$$
  $$=_E \mathsf{checksign}(T_1, \mathsf{vk}(T_2'), \mathsf{sign}(T_1, T_2'))\downarrow$$
  $$=_E \mathsf{ok}.$$

  Thus, $(M|_p =_E \mathsf{ok})\varphi_1$ and using $\varphi_1 \approx_s \varphi_2$, we have $(M|_p =_E \mathsf{ok})\varphi_2$ which also implies by extension $(M|_p =_E \mathsf{ok})\overline{\varphi}_2$ and we conclude.

  – $T_3 = x$. Then, we have $T_2\sigma_1 =_{AC} \mathsf{vk}(a)$, i.e. $T_2 =_{AC} \mathsf{vk}(a)$, which is $(M_2\varphi_1)\downarrow =_{AC} \mathsf{vk}(a)$. Thus, we have $(M_2 =_E \mathsf{idp_a})\varphi_1$ and, using $\varphi_1 \approx_s \varphi_2$ that leads us to $(M_2 =_E \mathsf{idp_a})\overline{\varphi}_2$. Moreover, $(M_1\overline{\varphi}_1)\downarrow =_{AC} T_1\sigma_1 =_{AC} (U'\varphi_1)\downarrow$, i.e. $T_1 =_{AC} (U'\varphi_1)\downarrow$ according to the definition of σ_1, which implies $(M_1\varphi_1)\downarrow =_{AC} (U'\varphi_1)\downarrow$. Using $\varphi_1 \approx_s \varphi_2$ again, we have $(M_1 =_E U')\overline{\varphi}_2$. Then:

  $$(M|_p\overline{\varphi}_2)\downarrow =_E \mathsf{checksign}((M_1\overline{\varphi}_2)\downarrow, (M_2\overline{\varphi}_2)\downarrow, (x\overline{\varphi}_2)\downarrow)\downarrow$$
  $$=_E \mathsf{checksign}((U'\varphi_2)\downarrow, (\mathsf{idp_a}\varphi_2)\downarrow, (U\varphi_2)\downarrow)\downarrow$$
  $$=_E \mathsf{ok}.$$

  And we conclude.

We now prove that the two claims hold.

Proof of Claim 1. We prove this Claim by induction on the size of M. We suppose w.l.o.g. that M is in normal form.

Base case: $M = y$ is a variable (or a name). Since φ_i is in normal form, we have that for all T s.t. $M\varphi_i \longrightarrow_{dt}^* T$ then $M\varphi_i =_{AC} T$. If M is a name, result is straightforward. If M is a variable, then $(U\varphi_i)\!\downarrow\ \in \mathsf{St}(T)$ implies that $(U\varphi_i)\!\downarrow\ \in \mathsf{St}(\varphi_i)$ which is a contradiction.

Induction Step: We assume that for any term M of size m s.t. $\mathsf{fn}(M) \cap \omega_i = \emptyset$ we have, for all term T s.t. $M\varphi_i \longrightarrow_{dt}^* T$, $(U\varphi_i)\!\downarrow\ \notin \mathsf{St}(T)$. We now consider a term M of size $(m+1)$, i.e. $M = \mathsf{f}(M_1, \ldots, M_n)$ with M_i of size at most equal to m. Moreover, following a down-to-top reduction strategy, we have $M\varphi_i = \mathsf{f}(M_1\varphi_i, \ldots, M_n\varphi_i) \longrightarrow_{dt}^* \mathsf{f}(T_1, \ldots, T_n)$ with $T_j = (M_j\varphi_i)\!\downarrow$. According to the induction hypothesis, we now that for all term T s.t. $M_j\varphi_i \longrightarrow_{dt}^* T$, then $(U\varphi_i)\!\downarrow\ \notin \mathsf{St}(T)$. Let us consider now the down-to-top reduction strategy $M\varphi_i \longrightarrow_{dt}^* T = \mathsf{f}(T_1', \ldots, T_n') \longrightarrow_{dt}^* \mathsf{f}(T_1, \ldots, T_n)$. If $(U\varphi_i)\!\downarrow\in \mathsf{St}(T)$, then we have two possibilities. Either $(U\varphi_i)\!\downarrow\in \mathsf{St}(T_k')$ for some $k \in [\![1, n]\!]$ and we have a contradiction with the induction hypothesis; or $\mathsf{f}(T_1', \ldots, T_n') =_{AC} (U\varphi_i)\!\downarrow$ and $\mathsf{f} = \mathsf{sign}$, $n = 2$ and $T_2' = \mathsf{a}$ which implies that a is deducible by φ_i, contradiction. Thus, $(U\varphi_i)\!\downarrow$ does not occur in any of the terms in the reduction $(M\varphi_i)\!\downarrow\longrightarrow_{dt}^* \mathsf{f}(T_1, \ldots, T_n)$. We now consider the reduction of $\mathsf{f}(T_1, \ldots, T_n)$:

- It is in normal form, then, we have two subcases:

    - $\mathsf{f} \neq \mathsf{sign}$. Then, since $(U\varphi_i)\!\downarrow\notin \mathsf{St}(T_j)$ for $j \in [\![1, n]\!]$ by the induction hypothesis, we conclude.
    - $\mathsf{f} = \mathsf{sign}$. Using the induction hypothesis, if $(U\varphi_i)\!\downarrow\in \mathsf{St}(\mathsf{sign}(T_1, T_2))$, we must have $\mathsf{sign}(T_1, T_2) =_{AC} (U\varphi_i)\!\downarrow$ and a would be deducible by φ_i which is a contradiction.

- $\mathsf{f}(T_1, \ldots, T_n)$ is not in normal form. Then, we have $\mathsf{f} \neq \mathsf{sign}$ since there are no rule in E s.t. sign is in head. We have $\mathsf{f}(T_1, \ldots, T_n) \longrightarrow_{dt} T$ and we consider two subcases:

    - $T \in \mathsf{St}(T_1, \ldots, T_n)$, then T is in normal form and, using the induction hypothesis, we know that $(U\varphi_i)\!\downarrow\notin \mathsf{St}(T)$ and we conclude.
    - $T \notin \mathsf{St}(T_1, \ldots, T_n)$, then, according to E, we have used rule (4), (5), (6) or (8), (9), (10). The three last rules lead to $T = \mathsf{ok}$ which concludes straightforwardly and the three first rules lead to T in normal form with the property that if $(U\varphi_i)\!\downarrow\in \mathsf{St}(T)$, then $(U\varphi_i)\!\downarrow\in \bigcup \mathsf{St}(T_k)$, which would be a contradiction.

Proof of Claim 2. Let us prove this by induction on the number of reduction steps in $M\sigma \longrightarrow_{dt}^* T$.

Base case: If $M\sigma =_{AC} T$, the result is straightforward. If $M\sigma \longrightarrow_{dt} T$ in one step, then, there is a position p, a substitution θ and a rule $l \longrightarrow r$ (different from (8)) s.t. $(M\sigma)|_p =_{AC} l\theta \longrightarrow_{dt} r\theta$. Since σ is in normal form, p must be a position of M. In particular, if we consider $M' = M[z]_p$, we have:

$$M\sigma =_{AC} (M[z]_p\sigma)[l\theta]_p =_{AC} (M'\sigma)[l\theta]_p \text{ and } M\sigma \longrightarrow_{dt} (M'\sigma)[r\theta]_p.$$

We note $\delta : (U\varphi_i)\!\downarrow\mapsto x$. Then, we have $(M\sigma)\delta =_{AC} ((M'\sigma)[l\theta]_p)\delta$. Hypothesis on M states that there is no position q such that $M|_q =_{AC} (U\varphi_i)\!\downarrow$. Therefore, there is no position q such that $M'|_q =_{AC} (U\varphi_i)\!\downarrow$ and the following equalities hold: $(M\sigma)\delta =_{AC} M(\sigma\delta) =_{AC} M$ and $(M'\sigma)\delta =_{AC} M'(\sigma\delta) =_{AC} M'$. Moreover, we know that $(l\theta) \notin \mathsf{St}((U\varphi_i)\!\downarrow)$ otherwise $(U\varphi_i)\!\downarrow$ would not be in normal form. Thus, $M =_{AC} M'[(l\theta)\delta]_p$. Since $l \longrightarrow r$ is different from rule (8), we have $(l\theta)\delta = l(\theta\delta) = l\theta' \longrightarrow r\theta'$, for $\theta' = \theta\delta$, and this implies $M \longrightarrow_{dt} M'[r\theta']_p$ following a down-to-top reduction strategy since if there was a lower reduction in M it would also exists in $M\sigma$. Now, we can conclude since:

$$
\begin{aligned}
(M'[r\theta']_p)\sigma &=_{AC} (M'\sigma)[(r\theta')\sigma]_p \\
&=_{AC} (M'\sigma)[r(\theta'\sigma)]_p \\
&=_{AC} (M'\sigma)[r\theta]_p.
\end{aligned}
$$

Induction Step: We suppose that, for any term M s.t. for all term T s.t. $M \longrightarrow_{dt}^* T$, then $(U\varphi_i)\!\downarrow\notin \mathsf{St}(T)$, we have: if $M\sigma \longrightarrow_{dt}^m T$ in m steps involving no (8) rule, then $M \longrightarrow_{dt}^m T'$ with $T'\sigma =_{AC} T$.

We consider now M s.t. $M\sigma \longrightarrow_{dt}^{m+1} T$ in $(m+1)$ steps with no rules (8) in the reduction path. Then $M\sigma \longrightarrow_{dt}^{m} M_1 \longrightarrow_{dt} T$. Using the induction hypothesis, we have that $M \longrightarrow_{dt}^{m} M_1'$ with $M_1'\sigma =_{AC} M_1$, thus $M_1'\sigma \longrightarrow_{dt} T$. But M' is verifying hypothesis of Claim 2, otherwise we would have a contradiction with the fact that M is satisfying them. Then, using Base Case, we have $M_1' \longrightarrow_{dt} T'$ with $T'\sigma =_{AC} T$. Thus, $M \longrightarrow_{dt}^{m} M_1' \longrightarrow_{dt} T'$ and $T'\sigma =_{AC} T$, which allows us to conclude.

$\square$

6 Static Equivalence of the Final Frame

This section is devoted to the proof of Proposition 1, that states static equivalence of the final frame obtained in the relation $\mathcal{R}$. This proof relies on several lemmas. Some of them have been presented in the previous section, other are stated here and proved in appendix.

6.1 Definitions and Useful Lemmas

We first start by introducing some notations that will be useful for the following lemmas. Some of these definitions may seem artificial but mainly come from the establishment of the relation $\mathcal{R}$, which is described in the Appendix Section F.

Definition 7. *We introduce (or remind) the following definitions:*

$$\tilde{\omega} = a_1, a_3, id_1, id_2, t_1, t_2, id_R,$$

$$\theta_{\text{init}} = \{{}^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid k = 1..n\} \mid \{{}^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R}\} \mid \{{}^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid k = 1..3\},$$

$$\theta_0 = \theta_{\text{init}} \mid \{{}^{\mathsf{penc}(v_k, r_k, g_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k, \mathsf{t}_k, v_k, \mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k, \mathsf{p}_k \rangle, \mathsf{id}_k)}/_{\mathsf{si}_k} \mid k = 1..2\},$$

$$\theta_k = \theta_{k-1} \mid \{{}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(M_k)), \mathsf{id}_R)}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_k), \mathsf{dec}(\Pi_2(M_k), \mathsf{a}_3))}/_{rec_k}\},$$

$$\theta_\delta = \theta_n \mid \{{}^{\mathsf{dec}(U_{\delta(k)}, \mathsf{a}_1)}/_{res_k} \mid k = 1..n\},$$

$$\sigma_{\mathsf{L}} = \{{}^{\mathsf{a}}/_{v_1}, {}^{\mathsf{b}}/_{v_2}\}, \ \sigma_{\mathsf{R}} = \{{}^{\mathsf{b}}/_{v_1}, {}^{\mathsf{a}}/_{v_2}\},$$

$$\hat{\sigma} = \{{}^{M_k\alpha}/_{x_k}, {}^{U_k\alpha}/_{d_k}, {}^{W_k\alpha}/_{hb_k} \mid k = 1..n\} \mid \{{}^{N_k\alpha}/_{x_b^k} \mid k = 1..2\},$$

$$\hat{\sigma}_i^j = \{{}^{M_k\alpha}/_{x_k} \mid k = 1..i\} \mid \{{}^{N_k\alpha}/_{x_b^k} \mid k = 1..\min(i, 2)\} \mid \{{}^{U_k\alpha}/_{d_k}, {}^{W_k\alpha}/_{hb_k} \mid k = 1..j\}.$$

where α is a substitution representing the intermediate outputs visible by an adversary controlling the corrupted Ballot Box (full definition given in Section A), and M_k, N_k, U_k, W_k are free terms satisfying some conditions defined in Section F. They represent the inputs submitted by the adversary during the different steps of the protocol.

The first lemma ensures that several secret datas (like encryption/decryption keys, voters' secret IDs, etc.) remain secret during the execution of the protocol.

Lemma 9. *For $i \in \{1, 2\}$, $j \in \{1, 2, 3\}$ and any free term U, if M is of one of the following forms: $a_j + U$, $t_i * U$, id_i or $p(id_i)$, then $\nu\tilde{\omega}.\theta_\delta \nvdash M$, ie. M is not deducible from the frame $\nu\tilde{\omega}.\theta_\delta$.*

We then prove that the different outputs of the (honest) Receipt Generator, that is, a signature and a receipt for each submitted ballot, do not bring any relevant information to the adversary that could help him to make a difference between two different executions of the protocol. This property is expressed by the fact that we can add these terms to the frame without breaking the static equivalence.

Lemma 10. *For $i \in [\![0, n]\!]$, we have: $\nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_{\mathsf{R}}$.*

Finally, we show that we can control the form of what is sent to the Decryption Device, assuming it is approved by the Auditor.

Lemma 11. *Let us consider $M_1, \ldots, M_n$ free terms s.t. $\forall i \in [\![1, n]\!]$, $\phi_{\mathsf{R}}(\mathsf{idp}_{i+1}, M_{i+1})\theta_i\hat{\sigma}_i^0\sigma_{\mathsf{L}} = \mathsf{ok}$ with $\mathsf{fv}(M_{i+1}) \subseteq \mathsf{dom}(\theta_i) \cup \mathsf{dom}(\hat{\sigma}_i^0) \setminus \{x_b^i\}$. We suppose that $\forall i \in [\![1, n]\!]$, $\nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_{\mathsf{R}}$. We also consider $U_1, \ldots, U_n$ and $W_1, \ldots, W_n$ be free terms such that:*

- $\mathsf{fv}(U_{k+1}) \subseteq \mathsf{dom}(\theta_n) \cup \{\mathsf{dom}(\hat{\sigma}_n^0)\} \cup \{d_1, \ldots, d_k\}$ *and* $\mathsf{fv}(W_{k+1}) \subseteq \mathsf{dom}(\theta_n) \cup \{\mathsf{dom}(\hat{\sigma}_n^k)\}$,

- $\phi_{\mathsf{A}}(\mathsf{hash}(\langle U_1, \ldots, U_n \rangle), hb_r^1, \ldots, hb_r^n, W_1, \ldots, W_n)\theta_n\hat{\sigma}\sigma_{\mathsf{L}} = \mathsf{ok}$, *with the corresponding notation* $hb_r^i = \langle \mathsf{idp}_i, \mathsf{hash}(\Pi_1(M_i)) \rangle$.

Then, we have, for $\sigma \in \{\sigma_{\mathsf{L}}, \sigma_{\mathsf{R}}\}, \forall\, 1 \leq i \leq n$:

$$U_i\theta_n\hat{\sigma}\sigma =_E \mathsf{e}_i\theta_n\hat{\sigma}\sigma, \text{ for } i = 1, 2.$$

Moreover, there exist free terms U_1', U_2' *and free term* U_3' *or* $U_3' = \mathsf{pk}(\mathsf{a_k} + U)$ *with free term* U:

$$U_i\theta_n\hat{\sigma}\sigma =_E \mathsf{penc}(U_1', U_2', U_3')\theta_n\hat{\sigma}\sigma, \text{ for } i \in [\![3, n]\!].$$

6.2 Proving Proposition 1

With all these lemmas, we can now prove the final static equivalence, which is stated in Proposition 1.

Proposition 1. *Let* δ *is a substitution of* $[\![1, n]\!]$ *and* $^t\delta = \delta \circ [1 \mapsto 2, 2 \mapsto 1]$. *Let* θ_δ *be the frame as defined above. Then we have:*

$$\nu\tilde{\omega}.\theta_\delta\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\theta_{^t\delta}\sigma_{\mathsf{R}}, \text{ with } \sigma_{\mathsf{L}} = \{^\mathsf{a}/_{\mathsf{v}_1}, ^\mathsf{b}/_{\mathsf{v}_2}\} \text{ and } \sigma_{\mathsf{R}} = \{^\mathsf{b}/_{\mathsf{v}_1}, ^\mathsf{a}/_{\mathsf{v}_2}\}.$$

Proof. We introduce the notation $\hat{\theta}_i^\delta = \theta_n \mid \{^{dec_{\delta(k)}}/_{res_k} \mid k \in [\![1, i]\!]\}$ with $dec_{\delta(i+1)} = \mathsf{dec}(d_{\delta(i+1)}, \mathsf{a}_1)$. We show by induction that:

$$\nu\tilde{\omega}.\hat{\theta}_i^\delta\hat{\sigma}\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\hat{\theta}_i^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}.$$

Base case. The base case is ensured by Lemma 10 which guarantees that $\nu\tilde{\omega}.\theta_n\hat{\sigma}_n^0\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\theta_n\hat{\sigma}_n^0\sigma_{\mathsf{R}}$ thus, by rewriting, we have $\nu\tilde{\omega}.\hat{\theta}_0^\delta\hat{\sigma}\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\hat{\theta}_0^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}$.

Induction step. Suppose that $\nu\tilde{\omega}.\hat{\theta}_i^\delta\hat{\sigma}\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\hat{\theta}_i^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}$ for some $i \geq 0$. Let us show that:

$$\nu\tilde{\omega}.\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\hat{\theta}_{i+1}^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}.$$

We have that $dec_{\delta(i+1)} = \mathsf{dec}(U_{\delta(i+1)}, \mathsf{a}_1)$ in the frame $\hat{\theta}_{i+1}^\delta\hat{\sigma}$. We distinguish cases according to the value of $\delta(i+1)$:

- If $\delta(i+1) = 1$, then, using Lemma 11 (since we are considering the decryption step of the protocol, ϕ_A must have been successful), we know that $U_k\theta_n\hat{\sigma}\sigma = \mathsf{e}_k\theta_n\hat{\sigma}\sigma$ for $k \in \{1, 2\}$ and any $\sigma \in \{\sigma_{\mathsf{L}}, \sigma_{\mathsf{R}}\}$. Thus, we have that $res_{i+1}\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}} =_E \mathsf{dec}(U_1, \mathsf{a}_1)\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}} =_E \mathsf{v}_1\sigma_{\mathsf{L}}$. But, we also have the following equalities $res_{i+1}\hat{\theta}_{i+1}^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}} =_E \mathsf{dec}(U_2, \mathsf{a}_1)\hat{\theta}_{i+1}^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}} =_E \mathsf{v}_2\sigma_{\mathsf{R}}$. According to the definition of σ_{L} and σ_{R}, we have $\mathsf{v}_1\sigma_{\mathsf{L}} =_E \mathsf{v}_2\sigma_{\mathsf{R}} =_E \mathsf{a}$. Then, using the induction hypothesis and Lemma 3 with the free term $U = \mathsf{a}$ and we conclude.

- If $\delta(i+1) = 2$, we adopt the same argument used in the previous case (replacing 1 by 2 and 2 by 1) and conclude using the induction hypothesis and Lemma 3 with the free term $U = \mathsf{b}$.

- If $\delta(i+1) = j \in [\![3, n]\!]$, then $^t\delta(i+1) = \delta(i+1)$ and we have $res_{i+1} = \mathsf{dec}(U_j, \mathsf{a}_1)$. According to Lemma 11, we have that $U_j\theta_n\hat{\sigma}\sigma = \mathsf{penc}(N, P, Q)\,\theta_n\hat{\sigma}\sigma$ for free terms N, P, Q and any $\sigma \in \{\sigma_{\mathsf{L}}, \sigma_{\mathsf{R}}\}$. Thus, $res_{i+1}\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}} =_E \mathsf{dec}(\mathsf{penc}(N, P, Q), \mathsf{a}_1)\theta_n\hat{\sigma}\sigma_{\mathsf{L}}$. We consider two subcases:

    - $Q\theta_n\hat{\sigma}\sigma_{\mathsf{L}} =_E \mathsf{pk}(\mathsf{a}_1)$. Then, $res_{i+1}\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}} =_E N\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}}$ with free N. And, in that case, we also have $Q\theta_n\hat{\sigma}\sigma_{\mathsf{R}} =_E \mathsf{pk}(\mathsf{a}_1)$ (using the induction hypothesis and the fact that $\mathsf{g}_1 = \mathsf{pk}(\mathsf{a}_1)$) and $res_{i+1}\hat{\theta}_{i+1}^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}} =_E N\hat{\theta}_{i+1}^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}$ too. Thus, we conclude using Lemma 3.
    - $Q\theta_n\hat{\sigma}\sigma_{\mathsf{L}} \neq_E \mathsf{pk}(\mathsf{a}_1)$. Then, there is no reduction of res_{i+1} (in both frames) and we have two possibilities. If there exists $j \in [\![1, i]\!]$ such that $res_j\hat{\theta}_j^\delta\hat{\sigma}\sigma_{\mathsf{L}} = res_{i+1}\hat{\theta}_{i+1}^\delta\hat{\sigma}\sigma_{\mathsf{L}}$, then we conclude using Lemma 3. If there is not, then since a_1 is restricted and not deducible (Lemma 9), we have that res_{i+1} is neither deducible itself, nor a subterm of $\hat{\theta}_i^\delta\hat{\sigma}\sigma_{\mathsf{L}}$ and $\hat{\theta}_i^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}$. Now since dec is a destructor in E, we use Lemma 7 to conclude.

Finally, we have that $\nu\tilde{\omega}.\hat{\theta}_n^\delta\hat{\sigma}\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\hat{\theta}_n^{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}$, that is, changing the notations:

$$\nu\tilde{\omega}.\theta_\delta\hat{\sigma}\sigma_{\mathsf{L}} \approx_s \nu\tilde{\omega}.\theta_{^t\delta}\hat{\sigma}\sigma_{\mathsf{R}}. \qquad \square$$

$$\text{fst}(\text{pair}(x, y)) = x \tag{E'-1}$$

$$\text{snd}(\text{pair}(x, y)) = y \tag{E'-2}$$

$$\text{dec}(\text{penc}(x, r, \text{pk}(k)), k) = x \tag{E'-3}$$

$$\text{dec}(\text{blind}(\text{penc}(x, r, \text{pk}(k)), b), k) = \text{blind}(x, b) \tag{E'-4}$$

$$\text{renc}(\text{penc}(x, r, \text{pk}(\text{a}_1)), \text{a}_2) = \text{penc}(x, r, \text{pk}(\text{a}_3)) \tag{E'-5}$$

$$\text{unblind}(\text{blind}(x, b), b) = x \tag{E'-6}$$

$$\text{checksign}(x, \text{vk}(id), \text{sign}(x, id)) = \text{ok} \tag{E'-7}$$

$$\text{checkpfk}_1(\text{vk}(id), ball, \text{pfk}_1(id, r, x, ball)) = \text{ok} \tag{E'-8}$$

$$\text{where } ball = \text{penc}(x, r, \text{a}_1)$$

$$\text{checkpfk}_2(\text{vk}(id), ball, bball, \text{pfk}_2(\text{vk}(id), \text{a}_2, b, ball, bball)) = \text{ok} \tag{E'-9}$$

$$\text{where } ball = \text{penc}(x, r, \text{a}_1) \text{ and } bball = \text{blind}(\text{penc}(x, r, \text{a}_3), b)$$

Figure 7: Equational theory E' used in ProVerif

7 Further Corruption Cases Using ProVerif

In order to study further corruption cases, we have used ProVerif, one of the only tools that can analyse equivalence properties in the context of security protocols. Since ProVerif does not handle associative and commutative (AC) symbols, we had to simplify the equational theory, yielding theory E' defined by the equations of Figure 7. The main idea behind E' is to remove associative and commutative symbols from E. All equations besides the AC equations are left unchanged except Equations (E-5) and (E-6). Equation (E-5) states that two encryptions can be combined, This can no longer be reflected in our ProVerif model. Equation (E-6) models re-encryption. To get rid of AC symbols, we instantiate it with the keys of the protocol (a_1, a_2 and a_3), preserving the behavior of the protocol, yielding Equation (E'-5). The fact that we can consider a simplified equational theory weakens the attacker model: some attacks relying on crafty combinations of the messages may be missed. But as shown by our study, this allows to analyse (automatically) more corruption scenario.

Since ProVerif is designed to prove equivalences between processes that differ only by terms, we also needed to use another tool, ProSwapper [28], to cope with the (non deterministic) shuffle done by the Decryption service. More precisely, we actually used their algorithm to compute directly a slightly modified process in our ProVerif specification.

Table 1: Results and Performances for Proving Ballot Secrecy using ProVerif

Corrupted Voters	0	1	2	3	4	5
Honest Players	~1s (✓)	~6s (✓)	~40s (✓)	~16m (✓)	~32h (✓)	>48h (?)
Corrupted B	~3s (✓)	~28s (✓)	~2m (✓)	~14m (✓)	~2h (✓)	>48h (?)
Corrupted R	~2s (✓)	~17s (✓)	~2m (✓)	~12m (✓)	~95m (✓)	~19h (✓)
Corrupted A	<1s (✓)	~4s (✓)	~16s (✓)	~1m (✓)	~5m (✓)	~37m (✓)
Corr. R & Corr. A	<1s (✓)	~4s (✓)	~17s (✓)	~1m (✓)	~6m (✓)	~36m (✓)

The results[1] are displayed in Table 1 and the ProVerif files corresponding to our experimentation can be found at [35]. Our case study with ProVerif indicates that ballot secrecy is still preserved even when the Receipt generator is corrupted (as well as several voters), at least in the simplified theory E'.

---

[1]Tests made on a MacBook Pro (OSX El Capitan 10.11.6) i5 2,3 GHz with 4Go RAM, using the ProVerif 1.94p11 version.

8 Conclusion

We have proposed the first formal proof in a symbolic model that the e-voting protocol used in Norway indeed satisfies ballot secrecy, even when all but two voters are corrupted and even when the voters communications channels can be eavesdropped and when the Auditor and either the Ballot box or the Receipt generator are corrupted. As expected, ballot secrecy is no longer guaranteed if both the Ballot box and the Receipt generator are corrupted. Slightly more surprisingly, the protocol is not secure either if the Decryption service is corrupted or if the Ballot box and the Auditor are corrupted, as discussed in Section 4.3. Our results in Table 2. In this table, ✓ indicates that ballot secrecy is satisfied and × shows that there is an attack. In particular, all the attacks described in Section 4.3 are displayed in the table.

Table 2: Results for Ballot Secrecy

Corrupted Voters	0	5	n
Honest Players		✓ (Theorem 2)	
Corrupted B		✓ (Theorem 1)	
Corrupted R	✓ (ProVerif)		?
Corrupted A	✓ (ProVerif)		?
Corr. D + ⋆		× (Section 4.3)	
Corr. B & Corr. R		× (Section 4.3)	
Corr. B & Corr. A		× (Section 4.3)	
Corr. R & Corr. A	✓ (ProVerif)		?

Note: ⋆ stands for any other corrupted players (B, R, A) or none.

Instead of doing additional (long and technical) proofs, a further step consists in developing a procedure for automatically checking for equivalences. Of course, this is a difficult problem. A first decision procedure has been proposed in [16] but is limited to subterm convergent theories. An implementation has recently been proposed [15] but it does not support such a complex equational theory. An alternative step would be to develop a sound procedure that over-approximate the relation, losing completeness in the spirit of ProVerif [10] but tailored to privacy properties.

It is also important to note that the security of the protocol strongly relies on the way initial secrets are pre-distributed. For example, three private decryption keys a_1, a_2, a_3 (such that $a_1 + a_2 = a_3$) need to be securely distributed among (respectively) the Ballot box, the Receipt generator and the Decryption service. Also, a table (V, s_V) containing the blinding factor for each voter needs to be securely distributed to Ballot box and a table (V, d_V) containing a permutation for each voter needs to be securely distributed to the Receipt generator. Moreover, anyone with access with both the codes mailed to the voters and to the SMS emitted by the Receipt generator would immediately learn the values of all the votes. We did not find in the documentation how and by who all these secret values were distributed. It should certainly be clarified as it could be a weak point of the system.

It also remains to study other security properties such as receipt-freeness, coercion-resistance, and verifiability. Receipt-freeness seems to strongly rely on whether the voter may forge a fake table of receipts or fake the message received from the receipt generator. One important feature of the Norwegian protocol is to ensure verifiability even when the voter's computer is not trusted. Our formal model could serve as a basis, splitting further the voter's behavior from its computer.

References

[1] Martin Abadi and Cédric Fournet. Mobile Values, New names, and Secure Communication. In *28th ACM Symposium on Principles of Programming Languages (POPL)*, 2001.

[2] Myrto Arapinis, Sergiu Bursuc, and Mark Ryan. Reduction of equational theories for verification of trace equivalence: re-encryption and AC. In *First International Conference on Principles of Security and Trust (POST'12)*. Springer, 2012.

[3] Myrto Arapinis, Véronique Cortier, and Steve Kremer. When are three voters enough for privacy properties? In *Proceedings of the 21st European Symposium on Research in Computer Security (ESORICS'16)*, 2016.

[4] Alessandro Armando, David Basin, Yohan Boichut, Yannick Chevalier, Luca Compagna, Jorge Cuellar, Paul Hankes Drielsma, Pierre-Cyrille Héam, Olga Kouchnarenko, Jacopo Mantovani, Sebastian Mödersheim, David von Oheimb, Michaël Rusinowitch, Judson Santiago, Mathieu Turuani, Luca Viganò, and Laurent Vigneron. The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications. In *17th International Conference on Computer Aided Verification (CAV)*, 2005.

[5] David Basin, Jannik Dreier, and Ralf Sasse. Automated symbolic proofs of observational equivalence. In *ACM Conference on Computer and Communications Security (CCS'15)*, 2015.

[6] Josh Benaloh, Michael D. Byrne, Bryce Eakin, Philip T. Kortum, Neal McBurnett, Olivier Pereira, Philip B. Stark, Dan S. Wallach, Gail Fisher, Julian Montoya, Michelle Parker, and Michael Winn. Star-vote: A secure, transparent, auditable, and reliable voting system. In *2013 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections, (EVT/WOTE'13)*, 2013.

[7] David Bernhard, Véronique Cortier, Olivier Pereira, Ben Smyth, and Bogdan Warinschi. Adapting Helios for Provable Ballot Secrecy. In *16th European Symposium on Research in Computer Security (ESORICS)*, 2011.

[8] David Bernhard, Olivier Pereira, and Bogdan Warinschi. How Not to Prove Yourself: Pitfalls of the Fiat-Shamir Heuristic and Applications to Helios. In *International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, 2012.

[9] Karthikeyan Bhargavan, Ricardo Corin, Cédric Fournet, and Eugen Zalinescu. Cryptographically Verified Implementations for TLS. In *15th ACM Conference on Computer and Communications Security (CCS)*, 2008.

[10] Bruno Blanchet. An Automatic Security Protocol Verifier based on Resolution Theorem Proving (invited tutorial). In *20th International Conference on Automated Deduction (CADE)*, 2005.

[11] Bruno Blanchet. *Vérification Automatique de Protocoles Cryptographiques: Modèle Formel et Modèle Calculatoire*. Mémoire d'habilitation à diriger des recherches, Université Paris-Dauphine, 2008.

[12] Bruno Blanchet, Martín Abadi, and Cédric Fournet. Automated Verification of Selected Equivalences for Security Protocols. *Journal of Logical and Algebraic Methods in Programming*, 2008.

[13] Rohit Chadha, Ştefan Ciobâcă, and Steve Kremer. Automated verification of equivalence properties of cryptographic protocols. In *21th European Symposium on Programming (ESOP)*, 2012.

[14] David Chaum, Richard Carback, Jeremy Clark, Aleksander Essex, Stefan Popoveniuc, Ronald L. Rivest, Peter Y. A. Ryan, Emily Shen, Alan T. Sherman, and Poorvi L. Vora. Corrections to scantegrity II: end-to-end verifiability by voters of optical scan elections through confirmation codes. *IEEE Trans. Information Forensics and Security*, 5(1):194, 2010.

[15] Vincent Cheval, Hubert Comon-Lundh, and Stéphanie Delaune. Trace Equivalence Decision: Negative Tests and Non-determinism. In *18th ACM Conference on Computer and Communications Security (CCS)*, 2011.

[16] Véronique Cortier and Stéphanie Delaune. A Method for Proving Observational Equivalence. In *22nd Computer Security Foundations Symposium (CSF)*, 2009.

[17] Véronique Cortier and Ben Smyth. Attacking and fixing Helios: An analysis of ballot secrecy. *Journal of Computer Security*, 2013.

[18] Véronique Cortier and Cyrille Wiedling. A formal analysis of the Norwegian E-voting protocol. In *First International Conference on Principles of Security and Trust (POST)*, 2012.

[19] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. A Secure and Optimally Efficient Multi-Authority Election Scheme. In *International Conference on the Theory and Application of Cryptographic Techniques (EuroCrypt)*, 1997.

[20] Cas Cremers. The Scyther Tool: Verification, falsification, and analysis of security protocols. In *20th International Conference of Computer Aided Verification (CAV)*, 2008.

[21] Stéphanie Delaune, Steve Kremer, and Mark Ryan. Verifying Privacy-type Properties of Electronic Voting Protocols. *Journal of Computer Security*, 2009.

[22] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. A Practical Secret Voting Scheme for Large Scale Elections. In *Workshop on the Theory and Application of Cryptographic Techniques: Advances in Cryptology*, 1992.

[23] Kristian Gjøsteen. Analysis of an Internet Voting Protocol. Cryptology ePrint Archive, Report 2010/380, 2010.

[24] Kristian Gjøsteen. The norwegian internet voting protocol. Cryptology ePrint Archive, Report 2013/473, 2013.

[25] Norwegian Government. Article of the Norwegian Government on the Deployment of E-voting (last seen: 08-26-16). `https://www.regjeringen.no/en/aktuelt/Internet-voting-pilot-to-be-discontinued/id764300/`.

[26] Alex Halderman and Vanessa Teague. The New South Wales iVote System: Security Failures and Verification Flaws in a Live Online Election. In *5th International Conference on E-voting and Identity (VoteID'15)*, 2015.

[27] Dalia Khader and Peter Y. A. Ryan. Receipt freeness of prêt à voter provably secure. *IACR Cryptology ePrint Archive*, 2011:594, 2011.

[28] Petr Klus, Ben Smyth, and Mark D. Ryan. ProSwapper: Improved equivalence verifier for ProVerif (last seen: 08-26-16). `http://www.bensmyth.com/proswapper.php`, 2010.

[29] Ralf Küsters and Tomasz Truderung. An Epistemic Approach to Coercion-Resistance for Electronic Voting Protocols. In *IEEE Symposium on Security and Privacy (S&P))*, 2009.

[30] Byoungcheon Lee, Colin Boyd, Ed Dawson, Kwangjo Kim, Jeongmo Yang, and Seungjae Yoo. Providing Receipt-Freeness in Mixnet-Based Voting Protocols. In *6th International Conference on Information Security and Cryptology (ICISC)*, 2004.

[31] Jia Liu. A Proof of Coincidence of Labeled Bisimilarity and Observational Equivalence in Applied Pi Calculus. `http://lcs.ios.ac.cn/~jliu/papers/LiuJia0608.pdf`, 2011.

[32] Miyako Ohkubo, Fumiaki Miura, Masayuki Abe, Atsushi Fujioka, and Tatsuaki Okamoto. An Improvement on a Practical Secret Voting Scheme. In *2nd International Information Security Workshop (ISW)*, 1999.

[33] Alan T. Sherman, Richard Carback, David Chaum, Jeremy Clark, Aleksander Essex, Paul S. Herrnson, Travis Mayberry, Stefan Popoveniuc, Ronald L. Rivest, Emily Shen, Bimal Sinha, and Poorvi L. Vora. Scantegrity mock election at takoma park. In *4th International Conference on Electronic Voting 2010, (EVOTE'10)*, pages 45–61, 2010.

[34] Alwen Tiu and Jeremy E. Dawson. Automating Open Bisimulation Checking for the Spi Calculus. In *3rd IEEE Computer Security Foundations Symposium (CSF)*, 2010.

[35] Cyrille Wiedling. Source files for proverif code. http://people.rennes.inria.fr/Cyrille.Wiedling/Resources/resources.html.

In the following appendix sections, we show the full demonstration of lemmas that are presented, without their proofs, in the core of the paper. We also present numerous lemmas that are useful to reach the final results, but are too specific to present a real interest for future uses. Finally, we define the relation $\mathcal{R}$ and give the proof of Proposition 2.

A   Starting Lemmas

Let us first recall some definitions:

$$\tilde{\omega} = \mathsf{a}_1, \mathsf{a}_3, \mathsf{id}_1, \mathsf{id}_2, \mathsf{t}_1, \mathsf{t}_2, \mathsf{id}_R,$$

$$\theta_\delta = \{^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid \mathsf{k} = 1..n\} \mid \{^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R}\} \mid \{^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid \mathsf{k} = 1..3\} \mid$$

$$\{^{\mathsf{penc}(\mathsf{v}_k,\mathsf{t}_k,\mathsf{g}_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k,\mathsf{t}_k,\mathsf{v}_k,\mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k,\mathsf{p}_k\rangle,\mathsf{id}_k)}/_{\mathsf{si}_k} \mid \mathsf{k} = 1..2\} \mid$$

$$\{^{\mathsf{sign}(\mathsf{hash}(\Pi_1(x_k)),\mathsf{id}_R)}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_k),\mathsf{dec}(\Pi_2(x_k),\mathsf{a}_3))}/_{rec_k} \mid \mathsf{k} = 1..n\} \mid$$

$$\{^{\mathsf{dec}(d_{\delta(k)},\mathsf{a}_1)}/_{res_k} \mid \mathsf{k} = 1..n\}.$$

We now show the proof of Lemma 9, which shows that several secret informations remains secret even after the execution of the protocol.

Lemma 9. *For* $\mathsf{i} \in \{1,2\}$, $\mathsf{j} \in \{1,2,3\}$ *and any free term* U, *if* M *is of one of the following forms:* $\mathsf{a}_\mathsf{j} + U$, $\mathsf{t}_\mathsf{i} * U$, id_i *or* $\mathsf{p}(\mathsf{id}_\mathsf{i})$, *then* $\nu\tilde{\omega}.\theta_\delta \nvdash M$, *ie.* M *is not deducible from the frame* $\nu\tilde{\omega}.\theta_\delta$.

Proof. We define four properties:

1. Let N be a deducible term such that $\exists p$ such that $N|_p = \mathsf{a}_\mathsf{i} + U$ for $\mathsf{i} \in \{1,3\}$ and any term U. There are two cases:

   - $U = \epsilon$ (with $\mathsf{a}_\mathsf{i} + \epsilon = \mathsf{a}_\mathsf{i}$) and $p = p'.2$ such that $N|_{p'} = \mathsf{f}(N', \mathsf{a}_\mathsf{i})$ with $\mathsf{f} \in \{\mathsf{dec}, \mathsf{renc}\}$.
   - $p = p'.1.q$ such that $N|_{p'} =_{AC} \mathsf{pk}(\mathsf{a}_\mathsf{i} + U')$ for some U' and $\forall q' < q$, $\mathsf{head}(N|_{p'.1.q'}) = +$.

2. Let N be a deducible term such that $\exists p$ such that $N|_p = \mathsf{t}_\mathsf{i} * U$ for $\mathsf{i} \in \{1,2\}$ and any term U. There are two cases :

   - $p = p'.2.q$ such that $N|_{p'} =_{AC} \mathsf{penc}(P_1, \mathsf{t}_\mathsf{i} * U, K_1)$ and $\forall q' < q$, $head(N|_{p'.2.q'}) = *$.
   - $p = p'.2.q$ such that $N|_{p'} =_{AC} \mathsf{pfk}_1(P_1, \mathsf{t}_\mathsf{i} * U, P_2, P_3)$ and $\forall q' < q$, $head(N|_{p'.2.q'}) = *$.

3. Let N be a deducible term such that $\exists p$ such that $N|_p = \mathsf{id}_\mathsf{i}$ for $\mathsf{i} \in \{1,2\}$. There are five cases:

   - $p = p'.1$ and $N|_{p'} = \mathsf{vk}(\mathsf{id}_\mathsf{i})$.
   - $p = p'.1$ and $N|_{p'} = \mathsf{p}(\mathsf{id}_\mathsf{i})$.
   - $p = p'.1$ and $N|_{p'} = \mathsf{s}(\mathsf{id}_\mathsf{i}i)$.
   - $p = p'.1$ and $N|_{p'} = \mathsf{pfk}_1(\mathsf{id}_\mathsf{i}, P_1, P_2, P_3)$.
   - $p = p'.2$ and $N|_{p'} = \mathsf{sign}(N', \mathsf{id}_\mathsf{i})$.

4. Let N be a deducible term such that $\exists p$ such that $N|_p = \mathsf{p}(\mathsf{id}_\mathsf{i})$ for $\mathsf{i} \in \{1,2\}$. Then, $p = p'.1$ and $N|_{p'} = \mathsf{d}(\mathsf{p}(\mathsf{id}_\mathsf{i}), M)$.

We can see that these properties imply the undeducability of $\mathsf{a}_\mathsf{i} + U$, $\mathsf{t}_\mathsf{i} * U$, id_i and $\mathsf{p}(\mathsf{id}_\mathsf{i})$. Let us prove these properties by induction on the number of steps needed to deduce N.

Base Case: All terms in the frame θ_δ verify these properties.

Induction Hypothesis: Let $M_1, \ldots, M_k$ be terms in normal form, deducible in $i - 1$ steps verifying the properties. We prove that $f(M_1, \ldots, M_k)\!\downarrow$ also verifies them. There are two main cases:

- If $f(M_1, \ldots, M_k)$ is in normal form, then it is obvious that the result is true.

- If it is not in normal form, then, since $M_1, \ldots, M_k$ are in normal form, the reduction occurs in head.

  - If the applied rule is different from (E-4), (E-5) and (E-6), then the result is a subterm of $M_1, \ldots, M_k$ (or ok) and it is verifying the properties, using the induction hypothesis.

  - If the rule (E-4) is applied, then $\mathsf{dec}(M_1, M_2) \longrightarrow \mathsf{blind}(T_1, T_2)$ with T_1 and T_2 subterms of M_1, M_2 and are verifying the properties by induction hypothesis. Then, $\mathsf{blind}(T_1, T_2)$ verify the properties.

  - If the rule (E-5) is applied, then $\circ(M_1, M_2) \longrightarrow \mathsf{penc}(T_1 \diamond T_2, R_1 * R_2, P)$ with T_1, T_2, R_1, R_2, P subterms of M_1, M_2, all verifying the properties by induction hypothesis. Thus, the result, in normal form, is verifying the properties too.

  - Finally, if the rule (E-6) is applied, then $\mathsf{renc}(M_1, M_2) \longrightarrow$ $\mathsf{penc}(T, R, \mathsf{pk}(P_1 + P_2))$ with T, R, P_1 and P_2 subterms of M_1 and M_2. Thanks to the induction hypothesis, we can see that the the result is satisfying the property. Indeed, whatever the case where a_i is a subterm of P_1 or P_2, $\exists\, P'$ such that $P_1 + P_2 =_{AC} \mathsf{a_i} + P'$.

$\square$

The next lemma is used to simplify the final frame obtained directly in the final step of the relation $\mathcal{R}$. We narrow it down, keeping an iff relation, to a smaller equivalence. We need the following notations:

$$\theta_\delta^b = \{{}^{\mathsf{vk(id_k)}}/_{\mathsf{idp_k}}, {}^{\mathsf{s(id_k)}}/_{\mathsf{s_k}} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{vk(id_R)}}/_{\mathsf{idp_R}}\} \mid \{{}^{\mathsf{pk(a_k)}}/_{\mathsf{g_k}} \mid \mathsf{k} = 1..3\} \mid$$
$$\{{}^{\mathsf{ball_k}}/_{ba_k}, {}^{\mathsf{ball_k}}/_{ba'_k}, {}^{\mathsf{ok}}/_{conf_k} \mid \mathsf{k} = 1..2\} \mid$$
$$\{{}^{\mathsf{ok}}/_{go_r^k}, {}^{\mathsf{ok}}/_{sy_k}, {}^{\mathsf{d(p(id_k),dec(\Pi_2(x_k),a_3))}}/_{x_r^k} \mid \mathsf{k} = 3..n\} \mid$$
$$\{{}^{\mathsf{sign(hash(\Pi_1(x_k)),id_R)}}/_{sr_k}, {}^{\mathsf{d(p(id_k),dec(\Pi_2(x_k),a_3))}}/_{rec_k} \mid \mathsf{k} = 1..n\} \mid$$
$$\{{}^{\langle \mathsf{idp_k,hash(\Pi_1(x_k))}\rangle}/_{hr_k}, {}^{\mathsf{dec(d_{\delta(k)},a_1)}}/_{res_k} \mid \mathsf{k} = 1..n\},$$
$$\sigma_\mathsf{L} = \{{}^{\mathsf{a}}/_{\mathsf{v_1}}, {}^{\mathsf{b}}/_{\mathsf{v_2}}\}, \; \sigma_\mathsf{R} = \{{}^{\mathsf{b}}/_{\mathsf{v_1}}, {}^{\mathsf{a}}/_{\mathsf{v_2}}\},$$

$$\tilde{\sigma} = \{{}^{M_\mathsf{k}}/_{x_k}, {}^{U_\mathsf{k}}/_{d_k}, {}^{W_\mathsf{k}}/_{hb_k} \mid \mathsf{k} = 1..n\} \mid \{{}^{N_\mathsf{k}}/_{x_b^k} \mid \mathsf{k} = 1..2\},$$
$$\hat{\sigma} = \{{}^{M_\mathsf{k}\alpha}/_{x_k}, {}^{U_\mathsf{k}\alpha}/_{d_k}, {}^{W_\mathsf{k}\alpha}/_{hb_k} \mid \mathsf{k} = 1..n\} \mid \{{}^{N_\mathsf{k}\alpha}/_{x_b^k} \mid \mathsf{k} = 1..2\} \text{ with}$$
$$\alpha = \{{}^{\langle \mathsf{idp_k,e_k,p_k,si_k}\rangle}/_{ba_k}, {}^{ba_k}/_{ba'_k}, {}^{\mathsf{ok}}/_{conf_k} \mid \mathsf{k} = 1..2\} \mid$$
$$\{{}^{\langle \mathsf{idp_k,hash(\Pi_1(x_k))}\rangle}/_{hr_k} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{ok}}/_{go_r^k}, {}^{\mathsf{ok}}/_{sy_k}, {}^{rec_k}/_{x_r^k} \mid \mathsf{k} = 3..n\}.$$

Lemma 12. $\nu\tilde{\omega}.\theta_\delta^b \tilde{\sigma}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{t\delta}^b \tilde{\sigma}\sigma_R \Longleftrightarrow \nu\tilde{\omega}.\theta_\delta \hat{\sigma}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{t\delta} \hat{\sigma}\sigma_R.$

Proof. The proof relies on two lemmas presented in the core of the article.

Lemma 2. *Let* $\tilde{n}, \tilde{m}$ *be names,* θ_1, θ_2 *substitutions and let us define* $\phi = \nu\tilde{n}. \left(\theta_1 \mid \{{}^{\langle U_1, U_2\rangle}/_x\}\right)$ *and* $\psi = \nu\tilde{m}. \left(\theta_2 \mid \{{}^{\langle V_1, V_2\rangle}/_x\}\right)$ *two frames with* U_1, U_2, V_1, V_2 *some terms. We have:*

$$\phi \approx_s \psi \Longleftrightarrow \nu\tilde{n}. \left(\theta_1 \mid \{{}^{U_1}/_{x_1}, {}^{U_2}/_{x_2}\}\right) \approx_s \nu\tilde{m}. \left(\theta_2 \cup \{{}^{V_1}/_{x_1}, {}^{V_2}/_{x_2}\}\right).$$

Lemma 3. *Let* $\tilde{n}, \tilde{m}$ *be names,* θ_1, θ_2 *substitutions, and* $\phi = \nu\tilde{n}.\theta_1$ *and* $\psi = \nu\tilde{m}.\theta_2$ *frames. Let* U *be a free term, we have:*

$$\phi \approx_s \psi \Longleftrightarrow \nu\tilde{n}. \left(\theta_1 \mid \{{}^{U\theta_1}/_x\}\right) \approx_s \nu\tilde{m}. \left(\theta_2 \cup \{{}^{U\theta_2}/_x\}\right).$$

We use Lemma 3 to remove the redundant information contained in x_r^p and ba'_p, Lemma 2 to expand ba_1 and ba_2, and, finally, Lemma 3 to remove the different ok stored in the frame (go_r^k, sy_k and $conf_k$) that does not influence static equivalence. Using Lemma 3, we also can remove hr_k since they are free terms.

$\square$

B Base Case for Static Equivalence

We consider the following frame:

$$\theta_0 = \{{}^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R}\} \mid \{{}^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid \mathsf{k} = 1..3\} \mid$$
$$\{{}^{\mathsf{penc}(\mathsf{v}_k,\mathsf{t}_k,\mathsf{g}_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k,\mathsf{t}_k,\mathsf{v}_k,\mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k,\mathsf{p}_k \rangle,\mathsf{id}_k)}/_{\mathsf{si}_k} \mid \mathsf{k} = 1..2\}.$$

Lemma 13. $\nu\tilde{\omega}.\theta_0\sigma_L \approx_s \nu\tilde{\omega}.\theta_0\sigma_R$.

Proof. Note that the adversary can arbitrarily combine ciphertexts from the frame with ciphertexts in the frame or freshly constructed ciphertexts, thus we enrich the frame θ_0 with any such combination of ciphertexts. Formally, for any $\alpha_1, \alpha_2 \in \mathbb{N}$ and free terms K, P, R, we define $C_{\alpha_1,\alpha_2,K,P,R}$ as follows :

$$\mathsf{penc}(P \overset{2}{\underset{i=1}{\diamond}} v_i^{\alpha_i}, R \overset{2}{\underset{i=1}{*}} r_i^{\alpha_i}, \mathsf{pk}(\mathsf{a}_1 + K)).$$

We define the extended frame $\widehat{\varphi}$ as follows:

$$\widehat{\varphi} = \nu\tilde{\omega}.(\theta_0 | \{{}^{C_{\alpha_1,\alpha_2,K,P,R}}/_{x_{\alpha_1,\alpha_2,K,P,R}}} \mid \alpha_1, \alpha_2 \in \mathbb{N}, \ \alpha_1 + \alpha_2 \neq 1$$
$$\text{and terms } K, P, R \text{ such that } (\mathsf{fn}(K) \cup \mathsf{fn}(P) \cup \mathsf{fn}(R)) \cap \tilde{\omega} = \emptyset$$
$$\text{and } \mathsf{fv}(K, P, R) \subseteq \mathsf{dom}(\varphi_0).\})$$
$$= \nu\tilde{\omega}.\widehat{\theta}.$$

Note that $C_{1,0,\epsilon,\epsilon,\epsilon}$ is equal to e_1 and $C_{0,1,\epsilon,\epsilon,\epsilon}$ is equal to e_2 that is why they are not added to the frame, we also have that $\widehat{\varphi}$ is infinite. Using Lemma 3, we know that, in order to prove Lemma 13, it is sufficient to show that: $\widehat{\varphi}\sigma_L \approx_s \widehat{\varphi}\sigma_R$. We introduce the following two claims.

Claim 1. *Let M, N be two terms such that $(\mathsf{fv}(M) \cup \mathsf{fv}(N)) \subseteq \mathsf{dom}(\widehat{\varphi})$ and $\mathsf{fn}(M, N) \cap \tilde{\omega} = \emptyset$. If $M\widehat{\varphi}\sigma =_{AC} N\widehat{\varphi}\sigma$, for some $\sigma \in \{\sigma_L, \sigma_R\}$, then $M\widehat{\varphi} =_{AC} N\widehat{\varphi}$.*

Claim 2. *Let M be a term such that $\mathsf{fv}(M) \subseteq \mathsf{dom}(\widehat{\varphi})$ and $\mathsf{fn}(M) \cap \tilde{\omega} = \emptyset$. If $M\widehat{\varphi}\sigma \longrightarrow U$ for some $\sigma \in \{\sigma_L, \sigma_R\}$, then there exists N such that $\mathsf{fn}(N) \cap \tilde{\omega} = \emptyset$, $U =_{AC} N\widehat{\varphi}\sigma$ and $M\widehat{\varphi}\sigma' \longrightarrow N\widehat{\varphi}\sigma'$ for any $\sigma' \in \{\sigma_L, \sigma_R\}$.*

The above claims allow us to conclude our proof. Let M, N be two terms such that $\mathsf{fn}(M, N) \cap \tilde{\omega} = \emptyset$, $\mathsf{fv}(M, N) \subseteq \mathsf{dom}(\widehat{\varphi})$ and $M\widehat{\varphi}\sigma_L =_E N\widehat{\varphi}\sigma_L$. Thus $(M\widehat{\varphi}\sigma_L)\downarrow =_{AC} (N\widehat{\varphi}\sigma_L)\downarrow$. Applying repeatedly Claim 2, we deduce that there exists M' such that $(M\widehat{\varphi}\sigma_L)\downarrow =_{AC} M'\widehat{\varphi}\sigma_L$ and $M\widehat{\varphi}\sigma_R \longrightarrow^* M'\widehat{\varphi}\sigma_R$. Similarly, there exists N' such that $(N\widehat{\varphi}\sigma_L)\downarrow =_{AC} N'\widehat{\varphi}\sigma_L$ and $N\widehat{\varphi}\sigma_R \longrightarrow^* N'\widehat{\varphi}\sigma_R$. From $M'\widehat{\varphi}\sigma_L =_{AC} N'\widehat{\varphi}\sigma_L$ and Claim 1, we deduce $M'\widehat{\varphi} =_{AC} N'\widehat{\varphi}$. Therefore, $M'\widehat{\varphi}\sigma_R =_{AC} N'\widehat{\varphi}\sigma_R$ and thus $M\widehat{\varphi}\sigma_R =_E N\widehat{\varphi}\sigma_R$.

Proof of Claim 1: Assume by contradiction that there exist M, N two terms such that $M\widehat{\varphi}\sigma =_{AC} N\widehat{\varphi}\sigma$, for some $\sigma \in \{\sigma_L, \sigma_R\}$ and $M\widehat{\varphi} \neq_{AC} N\widehat{\varphi}$. Consider M, N two minimal terms that satisfy this property.

- M and N are both variables or ok. If $M = \mathsf{ok}$, then $N = \mathsf{ok}$ since we can not have $M\widehat{\varphi}\sigma =_{AC} N\widehat{\varphi}\sigma$ with N variable and this case leads to a contradiction. Thus, we have $x\widehat{\varphi}\sigma =_{AC} y\widehat{\varphi}\sigma$ and $x\widehat{\varphi} \neq_{AC} y\widehat{\varphi}$ with $x, y \in \mathsf{dom}(\widehat{\varphi})$ and $x \neq y$. We have $\mathsf{head}(x\widehat{\varphi}\sigma) \in \{\mathsf{penc}, \mathsf{pfk}_1, \mathsf{pk}, \mathsf{s}, \mathsf{sign}, \mathsf{vk}\}$. But, by construction of $\widehat{\varphi}$, σ does not change randomnesses used in penc or in pfk_1 nor the secret ids in $\mathsf{pk}, \mathsf{s}, \mathsf{sign}$ or vk, thus, randomnesses or ids uniquely determine the variable, which implies $x = y$ and a contradiction.

- M is a variable or ok and $N = \mathsf{f}(N_1, \ldots, N_p)$. If $M = \mathsf{ok}$, since $N\widehat{\varphi}\sigma = \mathsf{f}(N_1\widehat{\varphi}\sigma, \ldots, N_p\widehat{\varphi}\sigma)$, we have a contradiction since $M\widehat{\varphi}\sigma \neq_{AC} N\widehat{\varphi}\sigma$ in any case. If M is a variable, we have $\mathsf{head}(M\widehat{\varphi}\sigma) \in \{\mathsf{penc}, \mathsf{pfk}_1, \mathsf{pk}, \mathsf{s}, \mathsf{sign}, \mathsf{vk}\}$. If $\mathsf{head}(M\widehat{\varphi}\sigma) \in \{\mathsf{penc}, \mathsf{pfk}_1, \mathsf{sign}\}$, then, according to Lemma 9, we have a contradiction upon deducibility of randomnesses $\mathsf{t}_1, \mathsf{t}_2$ and secret ids $\mathsf{id}_1, \mathsf{id}_2$. If $\mathsf{head}(M\widehat{\varphi}\sigma) \in$

34

$\{\mathsf{pk}, \mathsf{s}, \mathsf{vk}\}$ we can also exclude cases where $M \in \{\mathsf{idp}_1, \mathsf{idp}_2, \mathsf{s}_1, \mathsf{s}_2, \mathsf{idp}_R, \mathsf{g}_1, \mathsf{g}_3\}$ using again Lemma 9. Nevertheless, in all other cases, we have $N = \mathsf{f}(N_1)$ with $\mathsf{f} \in \{\mathsf{pk}, \mathsf{s}, \mathsf{vk}\}$ and since σ does not affect secret ids and secret key, we must have $M\widehat{\varphi} = N\widehat{\varphi}$. Contradiction.

- $M = \mathsf{f}(M_1, \ldots, M_p)$ and $N = \mathsf{g}(N_1, \ldots, N_q)$. We have that $M\widehat{\varphi}\sigma = \mathsf{f}(M_1\widehat{\varphi}\sigma, \ldots, M_p\widehat{\varphi}\sigma)$ $=_{AC} N\widehat{\varphi}\sigma = \mathsf{g}(N_1\widehat{\varphi}\sigma, \ldots, N_q\widehat{\varphi}\sigma)$. We must have $\mathsf{f} = \mathsf{g}$. If f is not an AC-symbol, i.e. in $\{\circ, +, \diamond, *\}$, we have immediately $p = q$ and there must be $i \in [\![1, p]\!]$ such that $M_i\widehat{\varphi} \neq_{AC} N_i\widehat{\varphi}$ but $M_i\widehat{\varphi}\sigma =_{AC} N_i\widehat{\varphi}\sigma$. In that case we will have a contradiction since M_i and N_i will be smaller than M and N. If f is an AC-symbol, e.g. $\circ$, assume we have $M = \overset{p}{\underset{i=1}{\circ}} M_i$, where $\mathsf{head}(M_i) \neq \circ$, and $N = \overset{q}{\underset{i=1}{\circ}} N_i$, where $\mathsf{head}(N_i) \neq \circ$. Then, we must have $p = q$ again and $i, j \in [\![1, p]\!]$ such that $M_i\widehat{\varphi} \neq_{AC} N_j\widehat{\varphi}$ but $M_i\widehat{\varphi}\sigma =_{AC} N_j\widehat{\varphi}\sigma$, which leads, again, to a contradiction.

Proof of Claim 2: This result is proved by inspection of the rewriting rules. More precisely, assume that $M\widehat{\varphi}\sigma \to U$ for some $\sigma \in \{\sigma_L, \sigma_R\}$. It means that there exists a rewriting rule $l \to r$, a substitution θ and a position p such that $M\widehat{\varphi}\sigma|_p =_{AC} l\theta$. p cannot occur below M since $\widehat{\varphi}\sigma$ is in normal form. If $M|_p = l\theta'$ for some θ' then we conclude that we can rewrite M as expected. The only interesting case is thus when $M|_p$ is not an instance of l but $M\widehat{\varphi}\sigma|_p$ is. By inspection of the rules, $l \to r$ can only correspond to one of the equations (E-3) to (E-10).

- The case of equations (E-3) or (E-4) is ruled out by the fact that $\mathsf{a}_1 + K$ with free K is not deducible from $\nu\widetilde{\omega}.\theta_\delta$ (and thus from $\nu\widetilde{\omega}.\theta_0$) according to Lemma 9 and that $\widehat{\varphi}$ is just an extension of θ_0 including only deducible terms from that one.

- For the case of Equation (E-5), we have several possibilities.

  - If $M|_p = \mathsf{penc}(M_1, M_2, M_3) \circ \mathsf{penc}(M_4, M_5, M_6)$, then since we must have $M_3\widehat{\varphi}\sigma =_{AC} M_6\widehat{\varphi}\sigma$ in order to have a reduction, we have, using Claim 1 that $M_3\widehat{\varphi} =_{AC} M_6\widehat{\varphi}$ then $M|_p\widehat{\varphi} =_E \mathsf{penc}(M_1 \diamond M_3, M_2 * M_4, M_3)\widehat{\varphi}$ and we conclude.

  - If $M|_p = x \circ \mathsf{penc}(M_1, M_2, M_3)$ with $x \in \mathsf{dom}(\widehat{\varphi})$, then we have that $x = y_{\alpha_1, \alpha_2, K, P, R}$ with some $\alpha_1, \alpha_2 \in \mathbb{N}$ and free terms K, P and R and we must have $\mathsf{pk}(\mathsf{a}_1 + K)\widehat{\varphi}\sigma =_{AC} M_3\widehat{\varphi}\sigma$. In that case, we have two cases: $M_3 = \mathsf{g}_1$ and $K\widehat{\varphi}\sigma =_{AC} \epsilon$ the null term or $M_3 = \mathsf{g}_3$ and $K\widehat{\varphi}\sigma =_{AC} \mathsf{a}_2$, otherwise we have a contradiction with the fact that a_1 is not deducible (Lemma 9). In both cases, we have $M|_p\widehat{\varphi} =_E y_{\alpha_1, \alpha_2, K, P \diamond M_1, R * M_2}\widehat{\varphi}$ and we conclude.

  - If $M|_p = x_1 \circ x_2$ with $x_1, x_2 \in \mathsf{dom}(\widehat{\varphi})$, then we have that $x_1 = y_{\alpha_1, \alpha_2, K, P, R}$ and $x_2 = y_{\alpha'_1, \alpha'_2, K', P', R'}$ with $\alpha_1, \alpha_2, \alpha'_1, \alpha'_2 \in \mathbb{N}$ and free terms K, P, R, K', P' and R'. (Note that we can note $\mathsf{e}_i = y_{\delta_{1i}, \delta_{2i}, \epsilon, \epsilon, \epsilon}$ with δ_{ij} the Kronecker symbol.) Since $M|_p\widehat{\varphi}\sigma$ is reducing, we must have $(\mathsf{a}_1 + K)\widehat{\varphi}\sigma =_{AC} (\mathsf{a}_1 + K')\widehat{\varphi}\sigma$ which implies $K\widehat{\varphi}\sigma =_{AC} K'\widehat{\varphi}\sigma$. Thus, using Claim 1, we have $K\widehat{\varphi} =_{AC} K'\widehat{\varphi}$. Then, we use that $M|_p\widehat{\varphi} =_E y_{\alpha_1 + \alpha'_1, \alpha_2 + \alpha'_2, K, P \diamond P', R * R'}\widehat{\varphi}$ to conclude.

- In the case of Equation (E-6), the two non-obvious cases are either the case when $M|_p = \mathsf{renc}(x, U)$ or $M|_p = \mathsf{renc}(\mathsf{penc}(M_1, M_2, x), U)$ with $x \in \mathsf{dom}(\widehat{\varphi})$ and U a free term such that $\mathsf{fn}(U) \cap \widetilde{\omega} = \emptyset$.

  - In the first case, we must have that $x = y_{\alpha_1, \alpha_2, K, P, R}$ for some $\alpha_1, \alpha_2 \in \mathbb{N}$ and free terms K, P and R (otherwise there will be no reduction). Then, we have that $M|_p\widehat{\varphi} =_E y_{\alpha_1, \alpha_2, K+U, P, R}\widehat{\varphi}$ which allows us to conclude.

  - In the second case, we have that $x \in \{\mathsf{g}_1, \mathsf{g}_2, \mathsf{g}_3\}$. If $x = \mathsf{g}_1$, we have $M|_p\widehat{\varphi} =_E y_{0,0,U,M_1,M_2}\widehat{\varphi}$ and we conclude. If $x = \mathsf{g}_2$, we have $M|_p\widehat{\varphi} =_E N\widehat{\varphi}$ with $N = \mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{a}_2 + U))$ and we conclude. Finally, if $x = \mathsf{g}_3$, we have $M|_p\widehat{\varphi} =_E y_{0,0,\mathsf{a}_2+U,M_1,M_2}\widehat{\varphi}$ and we conclude.

- For the case of Equation (E-7), the result is straightforward since there is nothing in $\widehat{\varphi}$ starting with the function symbol blind.

- In the case of Equation (E-8), we have $M = \mathsf{checksign}(M_1, M_2, M_3)$.

- If M_3 is a variable, then we must have $M_3 = \mathsf{si}_1$ (or si_2). In that case, $M_1 = \langle \mathsf{e}_1, \mathsf{p}_1 \rangle$ (or $\langle \mathsf{e}_2, \mathsf{p}_2 \rangle$) and $M_2 = \mathsf{idp}_1$ (or idp_2) otherwise there is no reduction since $\mathsf{t}_1, \mathsf{t}_2, \mathsf{id}_1$ and id_2 are not deducible according to Lemma 9. But then we have $M\widehat{\varphi} \to \mathsf{ok}$ and we conclude.

- If M_3 is not a variable, $M = \mathsf{checksign}(M_1, M_2, \mathsf{sign}(M_3, M_4))$. If M_2 is not a variable, then $M_2 = \mathsf{vk}(M_2')$ and since $M\widehat{\varphi}\sigma$ is reducing, we have $M_1\widehat{\varphi}\sigma =_{AC} M_3\widehat{\varphi}\sigma$ and $M_2'\widehat{\varphi}\sigma =_{AC} M_4\widehat{\varphi}\sigma$. Then, using Claim 1, we have $M_1\widehat{\varphi} =_{AC} M_3\widehat{\varphi}$ and $M_2'\widehat{\varphi} =_{AC} M_4\widehat{\varphi}$. But, then we have $M\widehat{\varphi} \to \mathsf{ok}$ and we can conclude. If M_2 is variable, then $M_2 = \mathsf{idp}_i$ with $i \in [\![3,n]\!]$ otherwise we have a contradiction with Lemma 9. Thus, we have $M_4\widehat{\varphi}\sigma =_{AC} \mathsf{id}_i$ and we conclude using the same argument as in the case where M_2 is not a variable.

- In the case of Equation (E-9), we have $M = \mathsf{checkpfk}_1(M_1, M_2, M_3)$.

  - If M_3 is a variable, then $M_3 = \mathsf{p}_1$ (or p_2) then, using Lemma 9, we must have that $M_1 = \mathsf{idp}_1$ (or idp_2) and $M_2 = \mathsf{e}_1$ (or e_2) but then, we have that $M\widehat{\varphi} \to \mathsf{ok}$ and we conclude.

  - If M_3 is not a variable, then $M_3 = \mathsf{pfk}_1(M_1', M_2', M_3', M_4')$. Since $M\widehat{\varphi}\sigma$ is reducing, we have $M_2\widehat{\varphi}\sigma =_{AC} M_4'\widehat{\varphi}\sigma$ and $M_1\widehat{\varphi}\sigma =_{AC} \mathsf{vk}(M_1')\widehat{\varphi}\sigma$. Using Claim 1, $M_2\widehat{\varphi} =_{AC} M_4'\widehat{\varphi}$ and $M_1\widehat{\varphi} =_{AC} \mathsf{vk}(M_1')\widehat{\varphi}$ which implies that $M\widehat{\varphi} \to \mathsf{ok}$ and, again, we conclude.

- Finally, case of Equation (E-10) with $M = \mathsf{checkpfk}_2(M_1, M_2, M_3, M_4)$. According to the definition of $\widehat{\varphi}$, M_4 can not be a variable. Then $M_4 = \mathsf{pfk}_2(M_1', M_2', M_3', M_4', M_5')$. Since $M\widehat{\varphi}\sigma$ is reducing, we must have that $M_1\widehat{\varphi}\sigma =_{AC} M_1'\widehat{\varphi}\sigma$, $M_2\widehat{\varphi}\sigma =_{AC} M_4'\widehat{\varphi}\sigma$ and $M_3\widehat{\varphi}\sigma =_{AC} M_5'\widehat{\varphi}\sigma$. Using Claim 1, these equalities holds for $\widehat{\varphi}$, but then, we conclude that $M\widehat{\varphi} \to \mathsf{ok}$ and we conclude.

$\square$

C Additional Lemmas

We remind the following notations:

$$\theta_{\mathsf{init}} = \{ {}^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid \mathsf{k} = 1..n \} \mid \{ {}^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R} \} \mid \{ {}^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid \mathsf{k} = 1..3 \},$$

$$\theta_0 = \theta_{\mathsf{init}} \mid \{ {}^{\mathsf{penc}(\mathsf{v}_k, \mathsf{t}_k, \mathsf{g}_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k, \mathsf{t}_k, \mathsf{v}_k, \mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k, \mathsf{p}_k \rangle, \mathsf{id}_k)}/_{\mathsf{si}_k} \mid \mathsf{k} = 1..2 \},$$

$$\theta_k = \theta_{k-1} \mid \{ {}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(M_k)), \mathsf{id}_R)}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_k), \mathsf{dec}(\Pi_2(M_k), \mathsf{a}_3))}/_{rec_k} \}.$$

Lemma 14. *Let $\phi = \nu\tilde{\omega}.\theta_n$ and $M = f(M_1, \dots, M_k)$ with $f \in \{\mathsf{vk}, \mathsf{penc}, \mathsf{pfk}_1, \mathsf{sign}\}$. Then, $(M\phi)\!\downarrow = f((M_1\phi)\!\downarrow, \dots, (M_k\phi)\!\downarrow)$.*

Proof. Let $M = f(M_1, \dots, M_k)$ for $(M\phi)\!\downarrow$. Since vk, penc, pfk_1 and sign do not appear in head in any rule of E, we must have $(M\phi)\!\downarrow = f((M_1\phi)\!\downarrow, \dots, (M_k\phi)\!\downarrow)$ and we conclude. $\square$

Lemma 15. *Let $\phi = \nu\tilde{\omega}.\theta_n$ and $M = f(M_1, \dots, M_k)$ a minimal recipe of $(M\phi)\!\downarrow$ with $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$. Then, $(M\phi)\!\downarrow = f((M_1\phi)\!\downarrow, \dots, (M_k\phi)\!\downarrow)$.*

Proof. Let us prove this by induction on the depth of M.

Base Case: $M = f(M_1, \dots, M_k)$ with $M_1, \dots, M_k$ variables or names.

- If $f \in \{\mathsf{fst}, \mathsf{snd}\}$, since $\nexists\, x \in \mathsf{dom}(\phi)$ such that $\mathsf{head}(x\phi) = \mathsf{pair}$, we conclude immediately.

- If $f = \mathsf{dec}$ and $M = \mathsf{dec}(M_1, M_2)$, then $M_1 \in \{\mathsf{e}_1, \mathsf{e}_2\}$ but if there is a reduction, it means that we have $M_2\phi = \mathsf{a}_1$ which is impossible as a_1 is not deducible by Lemma 9.

- If $f = \mathsf{unblind}$, since $\nexists\, x \in \mathsf{dom}(\phi)$ such that $\mathsf{head}(x\phi) = \mathsf{blind}$, we conclude immediately.

Induction Hypothesis: We suppose that $\forall$ term M with a depth ≥ 1 satisfies the property. Let $M = f(M_1, \dots, M_k)$ of depth equal to $n+1$ with M_i of depths $\leq n$. If $f((M_1\phi)\!\downarrow, \dots, (M_k\phi)\!\downarrow)$ is in normal form, we conclude easily. Suppose that $f((M_1\phi)\!\downarrow, \dots, (M_k\phi)\!\downarrow)$ is not in normal form.

- If $f \in \{\mathsf{fst}, \mathsf{snd}\}$ and $M = f(M_1)$. The case where M_1 is a variable is excluded. Then $M_1 = g(M_1', \ldots, M_k')$ with $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{pair}, \mathsf{snd}, \mathsf{unblind}\}$.

    - If $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using the induction hypothesis, we have $(M_1\phi){\downarrow} = g((M_1'\phi){\downarrow}, \ldots, (M_k'\phi){\downarrow})$ and, thus, $(M\phi){\downarrow} = f((M_1\phi){\downarrow})$.
    - If $g = \mathsf{pair}$ we have a contradiction with the minimality of M.

- If $f = \mathsf{dec}$ and $M = \mathsf{dec}(M_1, M_2)$. The case where M_1 is a variable is excluded (see Base case). Then $M_1 = g(M_1', \ldots, M_k')$ with $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{renc}, \mathsf{snd}, \mathsf{unblind}, \circ\}$.

    - If $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, we conclude thanks to the induction hypothesis.
    - If $g = \mathsf{penc}$, there is a contradiction with the minimality of M.
    - If $g = \mathsf{renc}$ and $M_1 = \mathsf{renc}(M_1', M_2')$. The case where M_1' is a variable is still excluded (see Base case). Then $M_1' = g_1(M_1'', \ldots, M_k'')$ with $g_1 \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{snd}, \mathsf{unblind}, \circ\}$ where all cases, except $g_1 = \circ$ lead to a contradiction using the induction hypothesis or minimality of M. If $g_1 = \circ$ and $M_1' = \circ(M_1'', M_2'')$, we can still exclude the case where M_1'' is a variable and then $M_1'' = g_2(M_1''', \ldots, M_k''')$ with $g_2 \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{renc}, \mathsf{snd}, \mathsf{unblind}\}$. Again, the only viable case is $g_2 = \mathsf{renc}$. Then, repeating this argument (a finite number of time since depths are finite) we have that $M_1 = \mathsf{renc}(\circ(\mathsf{renc}(\cdots \circ (n_1, n_2))))$ or $M_1 = \mathsf{renc}(\circ(\mathsf{renc}(\ldots \mathsf{renc}(n_1, n_2))))$ with n_1 and n_2 of length 1. But then we have a contradiction since n_1 and n_2 can not be names (then there are no reduction) or variables (see Base case).
    - Finally, the case when $g = \circ$. Using a similar argument close to the one for the previous case, we have that $M_1 = \circ(\mathsf{renc}(\cdots \circ (m_1, m_2)))$ or $M_1 = \circ(\mathsf{renc}(\ldots \mathsf{renc}(m_1, m_2)))$ with m_1 and m_2 of length 1 and we conclude in the same way.

- If $f = \mathsf{unblind}$ and $M = \mathsf{unblind}(M_1, M_2)$. The case where M_1 is a variable is excluded (see Base case). Then $M_1 = g(M_1', \ldots, M_k')$ with $g \in \{\mathsf{blind}, \mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$.

    - If $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, we conclude thanks to the induction hypothesis.
    - If $g = \mathsf{blind}$, there is a contradiction with the minimality of M.

$\square$

Lemma 16. *Let $\phi = \nu\tilde{\omega}.\theta_n$ and M a minimal recipe of $(M\phi){\downarrow} = f(M_1, \ldots, M_k)$ with $f \in \{\mathsf{vk}, \mathsf{pfk}_1, \mathsf{sign}\}$. Then M is a variable or $M = f(M_1', \ldots, M_k')$.*

Proof. Suppose, by contradiction, that M is not a variable and $M = f(M_1, \ldots, M_p)$ and s.t. $(M\phi){\downarrow} = g(N_1, \ldots, N_q)$ with $g \in \{\mathsf{pfk}_1, \mathsf{sign}, \mathsf{vk}\}$ and $f \neq g$. This would implies that $\exists\, l \to r$ a reduction rule and θ a substitution such that $\mathsf{head}(l\theta) = f$ and $\mathsf{head}(r\theta) = g$. Then, it must be the case that $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$. But then, we have a contradiction, according to Lemma 15. $\square$

D Shape of Valid Ballots

In this section, we show that we can derive a specific shape for ballots that pass the different tests during the protocol execution. We first define a measure for the length of terms and provide a definition for what we call *valid ballots*.

Definition 8. *We define $|.| : T(\Sigma, \mathcal{X}, \mathcal{N}) \to \mathbb{N}$ recursively as follows:*

- $|u| = 1$ *for u a constant or a variable,*

- $|f(u_1, \ldots, u_n)| = \sum |u_i|$ *if $f \in \{+, *, \circ, \diamond\}$.*

- $|f(u_1, \ldots, u_n)| = 1 + \sum |u_i|$ *otherwise.*

We also define $L : T(\Sigma, \mathcal{X}, \mathcal{N}) \to \mathbb{N} \times \mathbb{N}$ *which is defined as* $L(M) = (||M||, |M|)$ *where* $||.|| : T(\Sigma, \mathcal{X}, \mathcal{N}) \to \mathbb{N}$ *is defined as follows with* $\#_\circ(M)$ *the number of* $\circ$ *symbols in* M:

- $||u|| = 0$ *for* u *a constant or a variable,*

- $||f(u_1, \dots, u_n)|| = \begin{cases} \#_\circ(f(u_1, \dots, u_n)) + \sum ||u_i||, & f \in \{\mathsf{penc}, \mathsf{renc}\} \\[2mm] \sum ||u_i||, & \textit{otherwise.} \end{cases}$

Definition 9. *Let* $id \in \{\mathsf{id}_1, \dots, \mathsf{id}_n\}$. *A term* M *is said to be a* id-*valid ballot if* $\phi_\mathsf{B}(id, M)$ *is verified, equivalently:*

$$\begin{cases} M &= &\langle M_1, M_2, M_3, M_4 \rangle \\ M_1 &=_E &\mathsf{vk}(id) \\ \mathsf{checksign}(\langle M_2, M_3 \rangle, M_1, M_4) &=_E &\mathsf{ok} \\ \mathsf{checkpfk}_1(M_1, M_2, M_3) &=_E &\mathsf{ok} \end{cases}.$$

Then, we can give the general shape of the two first ballots submitted by honest voters on which we have some extra information. But we first recall some definitions:

$$\theta_{\mathsf{init}} = \{ {}^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid \mathsf{k} = 1..n \} \mid \{ {}^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R} \} \mid \{ {}^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid \mathsf{k} = 1..3 \},$$

$$\theta_0 = \theta_{\mathsf{init}} \mid \{ {}^{\mathsf{penc}(\mathsf{v}_k, \mathsf{t}_k, \mathsf{g}_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k, \mathsf{t}_k, \mathsf{v}_k, \mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k, \mathsf{p}_k \rangle, \mathsf{id}_k)}/_{\mathsf{si}_k} \mid \mathsf{k} = 1..2 \},$$

$$\theta_k = \theta_{k-1} \mid \{ {}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(M_k)), \mathsf{id}_R)}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_k), \mathsf{dec}(\Pi_2(M_k), \mathsf{a}_3))}/_{rec_k} \},$$

$$\hat{\sigma}_i^j = \{ {}^{M_k \alpha}/_{x_k} \mid \mathsf{k} = 1..i \} \mid \{ {}^{N_k \alpha}/_{x_b^k} \mid \mathsf{k} = 1..\min(i, 2) \} \mid \{ {}^{U_k \alpha}/_{d_k}, {}^{W_k \alpha}/_{hb_k} \mid \mathsf{k} = 1..j \}.$$

Lemma 17. *We consider* $i \in \{1, 2\}$ *and* M *a free term such that* $\mathsf{fv}(M) \subseteq \mathsf{dom}(\theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L)$ *and* $M \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L$ *be an* id_i-*valid ballot. We suppose that* $\nu \tilde{\omega}.\theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L \approx_s \nu \tilde{\omega}.\theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_R$. *Then, we have, for* $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma =_E \langle \mathsf{idp}_i, \mathsf{e}_i, \mathsf{p}_i, \mathsf{si}_i \rangle \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma.$$

Proof. Let $i \in \{1, 2\}$ and M a free term s.t. $\mathsf{fv}(M) \subseteq \mathsf{dom}(\theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L)$ and $M \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L$ is a id_i-valid ballot. Let M' minimal in size - according to the measure of length L defined in Definition 8 - such that: $M' \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L =_E M \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L$.

According to Definition 9, we have $M' \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L =_E \langle P_1, P_2, P_3, P_4 \rangle$.

- Let suppose that M' is a variable. Since $\nexists\, x \in \mathsf{dom}(\theta_{i-1})$ such that $x \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L$ is a id_i-valid ballot, there is a contradiction.

- Thus, $M' = f(M_1, \dots, M_n)$ with $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{pair}, \mathsf{snd}, \mathsf{unblind}\}$ since only equations (E-1), (E-2), (E-3) and (E-7) can lead to $\langle P_1, P_2, P_3, P_4 \rangle$.

  - If $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we have a contradiction with the fact that $\mathsf{head}(M' \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L) = \mathsf{pair}$.

  - If $f = \mathsf{pair}$, then $M' = \langle M_1, M'' \rangle$, with some free M_1. By repeating this reasoning, we get that $M' = \langle M_1, M_2, M_3, M_4 \rangle$, with some free M_i for $i \in [\![1, 4]\!]$. Thus we have :

    $$M' \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L =_E \langle M_1, M_2, M_3, M_4 \rangle \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L$$
    $$=_E M \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L.$$

Since $\nu \tilde{\omega}.\theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_L \approx_s \nu \tilde{\omega}.\theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma_R$, then, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma =_E \langle M_1, M_2, M_3, M_4 \rangle \theta_{i-1} \hat{\sigma}_{i-1}^0 \sigma.$$

Moreover, using again Definition 9, we have that:

$$M_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{vk}(\mathsf{id_i}),$$
$$M_3\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{pfk}_1(\mathsf{id_i}, N_1, N_2, M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L),$$
$$M_4\hat{\sigma}_{i-1}\sigma^0_{i-1}\sigma_L =_E \mathsf{sign}(\langle M_2, M_3\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L, \mathsf{id_i}).$$

According to Lemma 16, we know that M_1, M_3 and M_4 must be variables or constructed terms. But since, according Lemma 9, $\mathsf{id_i}$ is not deducible, M_1, M_3 and M_4 must be variables. In that case, we must have $M_1 = \mathsf{idp_i}$, $M_3 = \mathsf{p_i}$ and $M_4 = \mathsf{si_i}$. And, according to the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$ and for some free M_2:

$$M\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \langle\mathsf{idp_i}, M_2, \mathsf{p_i}, \mathsf{si_i}\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

Now, since $M_3 = \mathsf{p_i}$ and according to Definition 9, we must have that $M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{penc}(\mathsf{v_i}, \mathsf{t_i}, \mathsf{g_1})$ with M_2 free. Thus, M_2 is a variable or $M_2 = f(M'_1, \ldots, M'_k)$ with $f \in \{\circ, \mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{renc}, \mathsf{snd}, \mathsf{unblind}\}$:

- If $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we get a contradiction on the fact that $\mathsf{head}(M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L) = \mathsf{penc}$.

- If $M_2 = \mathsf{penc}(M'_1, M'_2, M'_3)$ with free terms M'_1, M'_2 and M'_3, then using Lemma 14 we must have $M'_2 = \mathsf{t_i}$ which is in contradiction with Lemma 9 and the fact that $\mathsf{t_i}$ is not deducible.

- If $M_2 = \mathsf{renc}(M'_1, M'_2)$ with M'_1 and M'_2 free terms. We must have $M'_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{penc}(M''_1, M''_2, \mathsf{pk}(M''_3))$ leading to the fact that $\mathsf{a_1} =_{AC} M''_3 + M'_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ where M'_2 is free which is a contradiction with Lemma 9.

- If $M_2 = \overset{p}{\underset{k=1}{\circ}} M'_k$ with M'_k free and $\mathsf{head}(M'_k) \neq \circ$ for $k \in [\![1, p]\!]$. To have a reduction, we must have, for $k \in [\![1, p]\!]$, that $M'_k\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{penc}(M^1_k, M^2_k, \mathsf{pk}(M^3_k))$, and since $\mathsf{e_1}$ and $\mathsf{e_2}$ are the only variables leading to a penc-headed term, we must have one M'_k which is not a variable, otherwise we would have $\mathsf{t_1}^q * \mathsf{t_2}^s = \mathsf{t_1}$ with $q + s = p \geq 2$. Due to the study of previous cases, we must have that $M'_k = \mathsf{penc}(M''_1, M''_2, M''_3)$ or $\mathsf{renc}(\mathsf{penc}(M''_1, M''_2, M''_3), M''_4)$, but this leads to $M''_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L * U = \mathsf{t_1}$ with free M''_2 which leads to a contradiction due to Lemma 9.

Thus, we have M_2 variable and $M_2 = \mathsf{e_1}$, which allows us to conclude the proof, using equivalence between the two frames. $\qquad\square$

We know can give the general shape of the ballots submitted by the intruder, under the condition they are accepted by the Ballot Box. Once again, let us remind the usual notations:

$$\theta_{\mathsf{init}} = \{\mathsf{vk(id_k)}/\mathsf{idp_k}, \mathsf{s(id_k)}/\mathsf{s_k} \mid \mathsf{k} = 1..n\} \mid \{\mathsf{vk(id_R)}/\mathsf{idp_R}\} \mid \{\mathsf{pk(a_k)}/\mathsf{g_k} \mid \mathsf{k} = 1..3\},$$
$$\theta_0 = \theta_{\mathsf{init}} \mid \{\mathsf{penc(v_k,t_k,g_1)}/\mathsf{e_k}, \mathsf{pfk_1(id_k,t_k,v_k,e_k)}/\mathsf{p_k}, \mathsf{sign(\langle e_k,p_k\rangle,id_k)}/\mathsf{si_k} \mid \mathsf{k} = 1..2\},$$
$$\theta_k = \theta_{k-1} \mid \{\mathsf{sign(hash(\Pi_1(M_k)),id_R)}/\mathsf{sr_k}, \mathsf{d(p(id_k),dec(\Pi_2(M_k),a_3))}/\mathsf{rec_k}\},$$
$$\hat{\sigma}^j_i = \{M_k\alpha/x_k \mid \mathsf{k} = 1..i\} \mid \{N_k\alpha/x^k_b \mid \mathsf{k} = 1..\min(\mathsf{i}, 2)\} \mid \{U_k\alpha/d_k, W_k\alpha/hb_k \mid \mathsf{k} = 1..j\}.$$

Lemma 18. *Using previous notations, let $i \in [\![3, n]\!]$ and M a free term s.t. $\mathsf{fv}(M) \subseteq \mathsf{dom}(\theta_{i-1}\hat{\sigma}^0_{i-1})$ and $M\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ is an id_i-valid ballot. We suppose that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$. Then, for $\sigma \in \{\sigma_L, \sigma_R\}$:*

- $M\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \langle M_1, M_2, M_3, M_4\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *with free* $M_1,\ldots,M_4$,

- $M_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{vk(id_i)}\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$,

- $M_3\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{pfk_1(id_i}, N_1, N_2, N_3)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *with free* $N_1, \ldots, N_3$,

- $M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(N_2, N_1, U)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *with free* U *or* $U =_{AC} \mathsf{pk}(\mathsf{a_p} + U')$ *with free* U' *and* $\mathsf{a_p} \in \{\mathsf{a_1}, \mathsf{a_3}\}$.

Proof. Let $i \in \{3, \ldots, n\}$. Let M such that $M\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L$ is a id_i-valid ballot. Let M' minimal in size - according to the measure of length L defined in Definition 8 - such that :

$$M'\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L =_E M\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L \tag{\dagger}$$

Using Definition 9, we have $M'\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L = \langle P_1, P_2, P_3, P_4 \rangle$.

- Let suppose that M_i' is a variable. Since $\nexists\, x \in \mathsf{dom}(\theta_{i-1})$ such that $x\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L$ is a id_i-valid ballot, there is a contradiction.

- Thus, $M' = f(M_1, \ldots, M_p)$ with $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{pair}, \mathsf{snd}, \mathsf{unblind}\}$ since only equations (E-1), (E-2), (E-3) and (E-7) can lead to $\langle P_1, P_2, P_3, P_4 \rangle$.

    - If $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we have a contradiction with the fact that $\mathsf{head}(M_i'\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L) = \mathsf{pair}$.

    - If $f = \mathsf{pair}$, then $M' = \langle M_1, M'' \rangle$, with some free M_1. By repeating this reasoning, we get that $M' = \langle M_1, M_2, M_3, M_4 \rangle$, with some free terms M_1, M_2, M_3 and M_4. Thus we have :

$$M'\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L =_E \langle M_1, M_2, M_3, M_4 \rangle\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L$$
$$=_E M\theta_{i-1}\sigma_{i-1}^0\sigma_L.$$

Since $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_R$, then, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M\theta_{i-1}\sigma_{i-1}^0\sigma =_E \langle M_1, M_2, M_3, M_4 \rangle\theta_{i-1}\sigma_{i-1}^0\sigma.$$

Definition 9 implies that $M_1\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L =_E \mathsf{vk}(\mathsf{id}_i)$. Since id_i is deducible (it is not restricted), we have here a public test and the equivalence is sufficient to show that $M_1\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma =_E \mathsf{vk}(\mathsf{id}_i)\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma$ for $\sigma \in \{\sigma_L, \sigma_R\}$.

Moreover, using Definition 9, we also have that $M_3\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L =_E \mathsf{pfk}_1(\mathsf{id}_i, P_1, P_2, P_3)$.

- M_3 cannot be a variable since there is no $x \in \mathsf{dom}(\theta_{i-1})$ s.t. $x\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L = \mathsf{pfk}_1(\mathsf{id}_i, P_1, P_2, P_3)$ with $i \geq 3$.

- Thus, $M_3 = f(N_1, \ldots, N_p)$ with $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{pfk}_1, \mathsf{snd}, \mathsf{unblind}\}$ since only equations (E-1), (E-2), (E-3) and (E-7) lead to $\mathsf{pfk}_1(\mathsf{id}_i, P_1, P_2, P_3)$.

    - If $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we have a contradiction with the fact that $\mathsf{head}(M_3\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L) = \mathsf{pfk}_1$.

    - If $f = \mathsf{pfk}_1$, then $M_3 = \mathsf{pfk}_1(\mathsf{id}_i, N_1, N_2, N_3)$, with some free N_1, N_2, N_3. Since $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_R$, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_3\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma =_E \mathsf{pfk}_1(\mathsf{id}_i, N_1, N_2, N_3)\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma.$$

Finally, Definition 9 gives us that, for some term U, we have:

$$M_2\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L =_E \mathsf{penc}(N_2\theta_{i-1}\sigma_{i-1}\sigma_L, N_1\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L, U).$$

- If M_2 is a variable, then $M_2 \in \{\mathsf{e}_1, \mathsf{e}_2\}$. In that case, we would have $N_1\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L =_E \mathsf{t}_1$ (or t_2) with free N_1 which would mean that t_1 (or t_2) is deducible which is in contradiction with Lemma 9.

- Thus, $M_2 = f(V_1, \ldots, V_p)$ with $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{renc}, \mathsf{snd}, \mathsf{unblind}, \circ\}$ since only equations (E-1) to (E-3) and (E-5) to (E-7) lead to $\mathsf{penc}(P_1, P_2, P_3)$.

    - If $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we have a contradiction with the fact that $\mathsf{head}(M_2\theta_{i-1}\hat{\sigma}_{i-1}^0\sigma_L) = \mathsf{penc}$.

– If $f = \mathsf{renc}$ i.e. $M_2 = \mathsf{renc}(V_1, V_2)$. If V_1 is variable, then $V_1 \in \{\mathsf{e_1}, \mathsf{e_2}\}$ and we would have a contradiction with Lemma 9 since we would have $N_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{t_i}$ and $\mathsf{t_i}$ would be deducible. Then $V_1 = g(V'_1, \ldots, V'_p)$ with $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{renc}, \mathsf{unblind}, \circ\}$ since $\mathsf{head}(V_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L) = \mathsf{penc}$.

* If $g \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we have a contradiction with the fact that $\mathsf{head}(V_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L) = \mathsf{penc}$.
* If $g = \mathsf{renc}$ and $V_1 = \mathsf{renc}(V'_1, V'_2)$, we have a contradiction with the minimality of M_2 since $\mathsf{renc}(V'_1, V'_2 + V_2)$ is a smaller recipe than $\mathsf{renc}(\mathsf{renc}(V'_1, V'_2), V_2)$ for L.
* If $g = \circ$ and $V_1 = V'_1 \circ V'_2$, we also have a contradiction with the minimality of M_2 since $\mathsf{renc}(V'_1, V_2) \circ \mathsf{renc}(V'_2, V_2)$ is a smaller recipe than $\mathsf{renc}(V'_1 \circ V'_2, V_2)$ according to the Definition 8 of the measure L. Indeed, since $M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ is reducing, we have $(V'_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L)\!\downarrow = \mathsf{penc}(U_1, U'_1, U''_1)$ and $(V'_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L)\!\downarrow = \mathsf{penc}(U_2, U'_2, U''_2)$ with $U''_1 = U''_2$ implying that the two recipes lead to the same term.
* If $g = \mathsf{penc}$ and $V_1 = \mathsf{penc}(V'_1, V'_2, V'_3)$. We have two cases :
  · If V'_3 is a variable, then $V'_3 \in \{\mathsf{g_1}, \mathsf{g_2}, \mathsf{g_3}\}$ and $M_2 = \mathsf{renc}(\mathsf{penc}(V'_1, V'_2, \mathsf{g_p}), V_2)$ with free terms V'_1, V'_2 and V_2. In that case, we have :

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{renc}(\mathsf{penc}(V'_1, V'_2, \mathsf{g_p}), V_2)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L (\ddagger)$$
$$=_E \mathsf{penc}(V'_1, V'_2, \mathsf{pk}(\mathsf{a_p} + V_2))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$$

Thanks to the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$ and $(\ddagger)$, we also have that:

$$M_2\theta_{i-1}\hat{\sigma}_{i-1}\sigma_R =_E \mathsf{renc}(\mathsf{penc}(V'_1, V'_2, \mathsf{g_p}), V_2)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$$
$$=_E \mathsf{penc}(V'_1, V'_2, \mathsf{pk}(\mathsf{a_p} + V_2))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$$

Then, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(V'_1, V'_2, \mathsf{pk}(\mathsf{a_p} + V_2)))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

Since
$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{penc}(N_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L, N_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L, U)$$

we have that $V'_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E N_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ and $V'_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E N_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$. Using the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$, these equalities hold with σ_R. Finally, for $\sigma \in \{\sigma_L, \sigma_R\}$ and free V_2:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(N_2, N_1, \mathsf{pk}(\mathsf{a_p} + V_2))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

(With $\mathsf{a_2}$ deducible, $\mathsf{pk}(\mathsf{a_2} + V_2)$ can be seen as a free U.)

  · If $V'_3 = h(V''_1, \ldots, V''_q)$ with $h \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{pk}, \mathsf{snd}, \mathsf{unblind}\}$, we conclude easily with a contradiction when $h \neq \mathsf{pk}$ using Lemma 15. If $h = \mathsf{pk}$ then there is contradiction with minimality of M_2 since $\mathsf{penc}(V'_1, V'_2, \mathsf{pk}(V''_1 + V_2))$ is smaller than $\mathsf{renc}(\mathsf{penc}(V'_1, V'_2, \mathsf{pk}(V''_3)), V_2)$.

– If $f = \mathsf{penc}$ i.e. $M_2 = \mathsf{penc}(V_1, V_2, V_3)$ with free terms V_1, V_2, V_3, we immediately conclude that, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(V_1, V_2, V_3)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

Using that $V_1\theta_{i-1}\hat{\sigma}^0_{i-1}\Sigma =_E N_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ and $V_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E N_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$, we have, with free V_3 and $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(N_2, N_1, V_3)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

– If $f = \circ$ i.e. $M_2 = V_1 \circ V_2$. Then, we can assume, without loss of generality, that $\mathsf{head}(V_1) \neq \circ$. Indeed, if $V_1 = V_1' \circ V_1''$, then we write $M_2 = V_1' \circ (V_1'' \circ V_2)$. Then, we have $M_2 = V_1 \circ \left(\circ_{j=2}^{p} V_j\right)$ s.t. $\forall\, j$, $\mathsf{head}(V_j) \neq \circ$. If $\exists\, j$ such that V_j is a variable, then $V_j \in \{e_1, e_2\}$ and $N_1 \theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L = r_k + U$ (where $k \in \{1,2\}$) with free N_1 which is a contradiction with Lemma 9 since $r_k + U$ is not deducible. Then, $\forall\, j \in [\![1,n]\!]$, $V_j = h_j(V_1^j, \ldots, V_q^j)$ with $h_j \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{penc}, \mathsf{renc}, \mathsf{unblind}\}$ since $\mathsf{head}(V_j\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L) = \mathsf{penc}$ otherwise $\mathsf{head}(V\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L) \neq \mathsf{penc}$. Now, according to Lemma 15, we have that $h_j \in \{\mathsf{penc}, \mathsf{renc}\}$ for $j \in [\![1,n]\!]$. Then:

$$M_2 = \left(\overset{p_1}{\underset{j=1}{\circ}}\, \mathsf{penc}(V_1^j, V_2^j, V_3^j)\right) \circ \left(\overset{p_2}{\underset{j=1}{\circ}}\, \mathsf{renc}(V_4^j, V_5^j)\right).$$

If $p_1 \geq 2$, then we have a contradiction with the minimality of M_2 since $\mathsf{penc}(V_1^1 \diamond V_1^2, V_2^1 * V_2^2, V_3^1)$ is a smaller recipe than $\mathsf{penc}(V_1^1, V_2^1, V_3^1) \circ \mathsf{penc}(V_1^2, V_2^2, V_3^2)$ according to Definition 8. Thus $p_1 \in \{0, 1\}$. Moreover, using the case $f = \mathsf{renc}$ we also must have that $V_4^j = \mathsf{penc}(W_1^j, W_2^j, W_3^j)$ with W_3^j a variable.

* Suppose that $p_1 = 1$, then $W_3^1 \theta_{i-1}\hat{\sigma}_{i-1}\sigma_L$ is deducible. In order for M_2 to be reduced, we must have the same keys in both W_3^1 and $\circ_{j=1}^{p_2} \mathsf{renc}(V_4^j, V_5^j)$. But since $V_4^j = \mathsf{penc}(W_1^j, W_2^j, W_3^j)$ with W_3^j a variable, then $W_3^j \in \{g_1, g_2, g_3\}$, and we must have $W_3^j = g_2$ otherwise this would imply that a_1 or a_3 is deducible which is contradiction with Lemma 9. Since $\forall j \in [\![1,n]\!]$ we have $W_3^j = a_2$, we also have that $\forall j \in [\![1,n]\!]$, $V_5^j = V_5^1$ (with V_5^1 the minimal recipe in all the V_5^j) otherwise we will not have $\mathsf{pk}(a_2 + V_5^j)\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L =_E \mathsf{pk}(a_2 + V_5^1)\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L$ and there will be no reduction. Then, finally, we have:

$$M_2 = \left(\overset{p_2}{\underset{j=1}{\circ}}\, \mathsf{renc}(\mathsf{penc}(W_1^j, W_2^j, g_2), V_5^1)\right)$$
$$\circ\, \mathsf{penc}(V_1^1, V_2^1, \mathsf{pk}(a_2 + V_5^1)).$$

Then, we have :
$$M_2\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L = \left[\circ_{j=1}^{p_2} \mathsf{renc}(\mathsf{penc}(W_1^j, W_2^j, g_2), V_5^1)\right.$$
$$\left.\circ\, \mathsf{penc}(V_1^1, V_2^1, \mathsf{pk}(a_2 + V_5^1))\right]\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L \; (\star)$$
$$=_E \mathsf{penc}(\diamond_{j=1}^{p_2} W_1^j \diamond V_1^1, *_{j=1}^{p_2} W_2^j * V_2^1, \mathsf{pk}(a_2 + V_5^1))\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L.$$

Thanks to the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_R$ and $(\star)$, we also have that:
$$M_2\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_R = \left[\circ_{j=1}^{p_2} \mathsf{renc}(\mathsf{penc}(W_1^j, W_2^j, g_2), V_5^1)\right.$$
$$\left.\circ\, \mathsf{penc}(V_1^1, V_2^1, \mathsf{pk}(a_2 + V_5^1))\right]\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_R \; (\star)$$
$$=_E \mathsf{penc}(\diamond_{j=1}^{p_2} W_1^j \diamond V_1^1, *_{j=1}^{p_2} W_2^j * V_2^1, \mathsf{pk}(a_2 + V_5^1))\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_R.$$

Then, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_2\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma =_E \mathsf{penc}(V_1', V_2', \mathsf{pk}(a_2 + V_3'))\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma.$$

* If $p_1 = 0$, we have $M = \circ_{j=1}^{p_2} \mathsf{renc}(\mathsf{penc}(W_1^j, W_2^j, W_3^j), V_5^j)$ with W_3^j variables. To have a reduction, we still need to have that $\forall p, q \in [\![1,n]\!]$ $\mathsf{pk}(a_p + V_5^p) = \mathsf{pk}(a_q + V_5^q)$ where $a_p, a_q \in \{a_1, a_2, a_3\}$. That leads us to $M_2 = \circ_{j=1}^{p_2} \mathsf{renc}(\mathsf{penc}(W_1^j, W_2^j, g_k), V_5^1)$ with $k \in \{1, 2, 3\}$ and free W_1^j, W_2^j, V_5^1 (V_5^1 is the minimal recipe among all V_5^j).

Then, we have :

$$M_2\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L$$
$$= \circ_{j=1}^{p_2} \mathsf{renc}(\mathsf{penc}(W_1^j, W_2^j, g_k), V_5^k)\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L \; (\dagger\dagger)$$
$$=_E \mathsf{penc}(\diamond_{j=1}^{p_2} W_1^j, *_{j=1}^{p_2} W_2^j, \mathsf{pk}(a_k + V_5^1))\theta_{i-1}\hat{\sigma}_{i-1}^0 \sigma_L.$$

Thanks to the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$ and (††), we also have that:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$$
$$= \diamond^{p_2}_{j=1}\mathsf{renc}(\mathsf{penc}(W^j_1, W^j_2, \mathsf{g_k}), V^1_5)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$$
$$=_E \mathsf{penc}(\diamond^{p_2}_{j=1}W^j_1, *^{p_2}_{j=1}W^j_2, \mathsf{pk}(\mathsf{a_k} + V^1_5))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R.$$

Then, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(V'_1, V'_2, \mathsf{pk}(\mathsf{a_k} + V'_3))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

Then, in both cases ($p_1 = 1$ or $p_1 = 0$), we have, for $\sigma \in \{\sigma_L, \sigma_R\}$, there is $i \in [\![1,3]\!]$ such that:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(V'_1, V'_2, \mathsf{pk}(\mathsf{a_k} + V'_3))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

Since $M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{penc}(N_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L, N_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L, U)$ then $V'_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E N_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ and $V'_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E N_1\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$. Using the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$, these equalities hold with σ_R. Finally, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$M_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(N_2, N_1, \mathsf{pk}(\mathsf{a_k} + V'_3))\theta_{i-1}\sigma^0_{i-1}\sigma.$$

Again, $\mathsf{pk}(\mathsf{a_2} + V'_3)$ can be seen as a free term U since $\mathsf{a_2}$ is deducible.

$\square$

The next lemma gives the general shape of a ballot submitted by the intruder provided the fact that it is accepted by the Receipt generator in the first execution and shows that it is also accepted in the second execution. As usual, let us remind the usual notations:

$$\theta_{\mathsf{init}} = \{{}^{\mathsf{vk}(\mathsf{id_k})}/_{\mathsf{idp_k}}, {}^{\mathsf{s}(\mathsf{id_k})}/_{\mathsf{s_k}} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{vk}(\mathsf{id_R})}/_{\mathsf{idp_R}}\} \mid \{{}^{\mathsf{pk}(\mathsf{a_k})}/_{\mathsf{g_k}} \mid \mathsf{k} = 1..3\},$$
$$\theta_0 = \theta_{\mathsf{init}} \mid \{{}^{\mathsf{penc}(\mathsf{v_k}, \mathsf{t_k}, \mathsf{g_1})}/_{\mathsf{e_k}}, {}^{\mathsf{pfk}_1(\mathsf{id_k}, \mathsf{t_k}, \mathsf{v_k}, \mathsf{e_k})}/_{\mathsf{p_k}}, {}^{\mathsf{sign}(\langle\mathsf{e_k}, \mathsf{p_k}\rangle, \mathsf{id_k})}/_{\mathsf{si_k}} \mid \mathsf{k} = 1..2\},$$
$$\theta_k = \theta_{k-1} \mid \{{}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(M_k)), \mathsf{id_R})}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id_k}), \mathsf{dec}(\Pi_2(M_k), \mathsf{a_3}))}/_{rec_k}\},$$
$$\hat{\sigma}^j_i = \{{}^{M_k\alpha}/_{x_k} \mid \mathsf{k} = 1..i\} \mid \{{}^{N_k\alpha}/_{x^k_b} \mid \mathsf{k} = 1..\min(i,2)\} \mid \{{}^{U_k\alpha}/_{d_k}, {}^{W_k\alpha}/_{hb_k} \mid \mathsf{k} = 1..j\}.$$

Lemma 19. *We consider $i \in [\![1, n]\!]$ and M a free term such that $\mathsf{fv}(M) \subseteq \mathsf{dom}(\theta_{i-1}\hat{\sigma}^0_{i-1})$ and such that $\phi_\mathsf{R}(\mathsf{idp_i}, M)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{ok}$ with:*

$$\phi_\mathsf{R}(id, x) = (x = \langle x_1, x_2, x_3\rangle) \wedge (x_1 = \langle y_1, y_2, y_3, y_4\rangle) \wedge (y_1 = id)$$
$$\wedge (\mathsf{checksign}(\langle y_2, y_3\rangle, y_1, y_4)) \wedge (\mathsf{checkpfk}_1(y_1, y_2, y_3))$$
$$\wedge (\mathsf{checkpfk}_2(y_1, y_2, x_2, x_3)).$$

Assuming that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$, then, for $\sigma \in \{\sigma_L, \sigma_R\}$:

- $\phi_\mathsf{R}(\mathsf{idp_i}, M)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{ok}$,

- $M\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \langle P, Q, R\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *for free terms P, Q and R.*

- *For $i \in \{1, 2\}$, we have:*

  - $P\theta_{i-1}\hat{\sigma}a^0_{i-1}\sigma =_E \langle\mathsf{idp_i}, \mathsf{e_i}, \mathsf{p_i}, \mathsf{si_i}\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$
  - $Q\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{blind}(\mathsf{renc}(\mathsf{e_i}, Q_1), Q_2)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *with Q_1 and Q_2 free terms.*

- *For $i \in [\![3, n]\!]$, we have:*

  - $P\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \langle P_1, P_2, P_3, P_4\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *with $P_1, \ldots, P_4$ free terms,*
  - $Q\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{blind}(\mathsf{penc}(Q_1, Q_2, U), Q_3)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ *with free Q_1, Q_2, Q_3, U or $U =_{AC}$ $\mathsf{pk}(\mathsf{a_p} + U')$ with free U' and $\mathsf{a_p} \in \{\mathsf{a_1}, \mathsf{a_3}\}$.*

Proof.

- Using the fact that $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$ and since $\phi_R(\mathsf{idp_i}, M)$ is a public test, we have that the first property holds obviously.

- Now, using that $\phi_R(\mathsf{idp_i}, M)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{ok}$, we have that, $M\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \langle M_1, M_2, M_3\rangle$. By repeating the same reasoning as we did in the proof of Lemma 18, we have that $M\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \langle P, Q, R\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ with free term P, Q and R for $\sigma \in \{\sigma_L, \sigma_R\}$.

- Let $i \in \{1, 2\}$.

  - Since $\phi_R(\mathsf{idp_i}, M)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{ok}$, we have that $P\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ must be an id_i-valid ballot. Then, we use Lemma 17 to conclude on the form of $P\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$.

  - We also have that $\mathsf{checkpfk_2}(\mathsf{idp_i}, \mathsf{e_i}, Q, R)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{ok}$, thus $R\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E \mathsf{pfk_2}(R_1, R_2, R_3, R_4, R_5)$ with $R_1 = \mathsf{vk}(\mathsf{id_i})$ and $R_4 = \mathsf{e_i}\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$. We consider R' minimal in size (for the measure L) such that $R\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L =_E R'\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$.

    * R' cannot be a variable since there is no variable $x \in \mathsf{dom}(\theta_{i-1})$ such that $\mathsf{head}(x\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L) = \mathsf{pfk_2}$.
    * $R' = f(R_1, \ldots, R_p)$ with $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{pfk_2}, \mathsf{snd}, \mathsf{unblind}\}$ since only equations (E-1), (E-2), (E-3) and (E-7) can lead to a term of the form $\mathsf{pfk_2}(R_1, R_2, R_3, R_4, R_5)$.
      · If $f \in \{\mathsf{dec}, \mathsf{fst}, \mathsf{snd}, \mathsf{unblind}\}$, using Lemma 15, we have a contradiction with the fact that $\mathsf{head}(R'\theta_{i-1}\hat{\sigma}a^0_{i-1}\sigma_L) = \mathsf{pfk_2}$.
      · If $f = \mathsf{pfk_2}$, then $R' = \mathsf{pfk_2}(R_1, R_2, R_3, R_4, R_5)$, with some free R_j for $j \in [\![1, 5]\!]$. Since we have that $(R =_E \mathsf{pfk_2}(\mathsf{idp_i}, R_1, R_2, \mathsf{e_i}, R_3))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ and $\nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L \approx_s \nu\tilde{\omega}.\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R$, we conclude that:

$$(R =_E \mathsf{pfk_2}(\mathsf{idp_i}, R_1, R_2, \mathsf{e_i}, R_3))\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_R.$$

    Thus, for $\sigma \in \{\sigma_L, \sigma_R\}$, we have:

$$R\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{pfk_2}(\mathsf{idp_i}, R_1, R_2, \mathsf{e_i}, R_3)\theta_{i-1}\hat{\sigma}a^0_{i-1}\sigma.$$

    Moreover, we also have, according to Equation E-10, and repeating the same development as we just did, that:

$$Q\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{blind}(\mathsf{renc}(\mathsf{e_i}, R_1), R_2)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma.$$

- Let $i \in [\![3, n]\!]$.

  - Since $\phi_R(\mathsf{idp_i}, M)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{ok}$, we have that $P\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma_L$ is an id_i-valid ballot. Using Lemma 18, we have $P\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \langle P_1, P_2, P_3, P_4\rangle\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ with free terms P_j for $j \in [\![1, 4]\!]$ and $\sigma \in \{\sigma_L, \sigma_R\}$.

  - According to Lemma 18, we also get that $P_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(P'_1, P'_2, U)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ for free terms P'_1, P'_2 and free term U or $U =_{AC} \mathsf{pk}(\mathsf{a_p} + U')$ with free U' and $\mathsf{a_p} \in \{\mathsf{a_1}, \mathsf{a_3}\}$. Now, according to Equation E-10 and by repeating the same reasoning as in the previous case in this situation, we can deduce that $R\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{pfk_2}(\mathsf{vk}(\mathsf{id_i}), R_1, R_2, R_3, R_4)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$ with free R_i for $i \in [\![1, 4]\!]$ and $\sigma \in \{\sigma_L, \sigma_R\}$. This equation also provides that $Q\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{blind}(\mathsf{renc}(P_2, R_1), R_2)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$. Now, since $P_2\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{penc}(P'_1, P'_2, U)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$, we can conclude that:

$$Q\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma =_E \mathsf{blind}(\mathsf{penc}(Q_1, Q_2, U), Q_3)\theta_{i-1}\hat{\sigma}^0_{i-1}\sigma$$

  with free Q_1, Q_2, Q_3 and free U or $U =_{AC} \mathsf{pk}(\mathsf{a_p} + U')$ with free U' and $\mathsf{a_p} \in \{\mathsf{a_1}, \mathsf{a_3}\}$.

$\square$

E   Moving to Final Static Equivalence

In this section, we present the proofs of the two last lemmas used for proving Proposition 1. We recall the different notations one last time:

$$\theta_{\mathsf{init}} = \{{}^{\mathsf{vk}(\mathsf{id}_\mathsf{k})}/_{\mathsf{idp}_\mathsf{k}}, {}^{\mathsf{s}(\mathsf{id}_\mathsf{k})}/_{\mathsf{s}_\mathsf{k}} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{vk}(\mathsf{id}_\mathsf{R})}/_{\mathsf{idp}_\mathsf{R}}\} \mid \{{}^{\mathsf{pk}(\mathsf{a}_\mathsf{k})}/_{\mathsf{g}_\mathsf{k}} \mid \mathsf{k} = 1..3\},$$

$$\theta_0 = \theta_{\mathsf{init}} \mid \{{}^{\mathsf{penc}(\mathsf{v}_\mathsf{k},\mathsf{t}_\mathsf{k},\mathsf{g}_1)}/_{\mathsf{e}_\mathsf{k}}, {}^{\mathsf{pfk}_1(\mathsf{id}_\mathsf{k},\mathsf{t}_\mathsf{k},\mathsf{v}_\mathsf{k},\mathsf{e}_\mathsf{k})}/_{\mathsf{p}_\mathsf{k}}, {}^{\mathsf{sign}(\langle \mathsf{e}_\mathsf{k},\mathsf{p}_\mathsf{k} \rangle,\mathsf{id}_\mathsf{k})}/_{\mathsf{si}_\mathsf{k}} \mid \mathsf{k} = 1..2\},$$

$$\theta_k = \theta_{k-1} \mid \{{}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(M_\mathsf{k})),\mathsf{id}_\mathsf{R})}/_{sr_\mathsf{k}}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_\mathsf{k}),\mathsf{dec}(\Pi_2(M_\mathsf{k}),\mathsf{a}_3))}/_{rec_\mathsf{k}}\},$$

$$\hat{\sigma}_i^j = \{{}^{M_\mathsf{k}\alpha}/_{x_\mathsf{k}} \mid \mathsf{k} = 1..i\} \mid \{{}^{N_\mathsf{k}\alpha}/_{x_b^\mathsf{k}} \mid \mathsf{k} = 1..\mathsf{min}(\mathsf{i},2)\} \mid \{{}^{U_\mathsf{k}\alpha}/_{d_\mathsf{k}}, {}^{W_\mathsf{k}\alpha}/_{hb_\mathsf{k}} \mid \mathsf{k} = 1..j\},$$

$$\sigma_\mathsf{L} = \{{}^{\mathsf{a}}/_{v_1}, {}^{\mathsf{b}}/_{v_2}\}, \ \sigma_\mathsf{R} = \{{}^{\mathsf{b}}/_{v_1}, {}^{\mathsf{a}}/_{v_2}\},$$

$$\hat{\sigma} = \{{}^{M_\mathsf{k}\alpha}/_{x_\mathsf{k}}, {}^{U_\mathsf{k}\alpha}/_{d_\mathsf{k}}, {}^{W_\mathsf{k}\alpha}/_{hb_\mathsf{k}} \mid \mathsf{k} = 1..n\} \mid \{{}^{N_\mathsf{k}\alpha}/_{x_b^\mathsf{k}} \mid \mathsf{k} = 1..2\} \text{ with}$$

$$\alpha = \{{}^{\langle \mathsf{idp}_\mathsf{k},\mathsf{e}_\mathsf{k},\mathsf{p}_\mathsf{k},\mathsf{si}_\mathsf{k} \rangle}/_{ba_\mathsf{k}}, {}^{ba_\mathsf{k}}/_{ba'_\mathsf{k}}, {}^{\mathsf{ok}}/_{conf_\mathsf{k}} \mid \mathsf{k} = 1..2\} \mid$$

$$\{{}^{\langle \mathsf{idp}_\mathsf{k},\mathsf{hash}(\Pi_1(x_\mathsf{k})) \rangle}/_{hr_\mathsf{k}} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{ok}}/_{go_r^\mathsf{k}}, {}^{\mathsf{ok}}/_{sy_\mathsf{k}}, {}^{rec_\mathsf{k}}/_{x_r^\mathsf{k}} \mid \mathsf{k} = 3..n\}.$$

We first show that we can add the signature coming from the Receipt generator in the frame.

Lemma 10. *For $i \in [\![0, n]\!]$, we have: $\nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_\mathsf{L} \approx_s \nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_\mathsf{R}$.*

Proof. The case $i = 0$ is proved by Lemma 13.

Induction Step: Assume now that, for any $i \geq 0$, we have that $\nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_L \approx_s \nu\tilde{\omega}.\theta_i\hat{\sigma}_i^0\sigma_R$ holds. We will show that it also hold for $i + 1$.

We consider the term $S_{i+1} = \mathsf{sign}(U_{i+1}, \mathsf{id}_\mathsf{R})$ with $U_{i+1} = \mathsf{hash}(\Pi_1(M_{i+1}))$ and free M_{i+1}. Since id_R is restricted and not deducible according to Lemma 9, we must have that $S_{i+1}\theta_i\hat{\sigma}_i^0\sigma$ does not appear in $\theta_i\hat{\sigma}_i^0\sigma$ for $\sigma \in \{\sigma_L, \sigma_R\}$. We also have that $\{{}^{\mathsf{vk}(\mathsf{id}_\mathsf{R})}/_{\mathsf{idp}_\mathsf{R}}\} \in \theta_i$, thus, using Lemma 8, we have, with $\theta_i^s = \theta_i \mid \{{}^{S_{i+1}}/_{sr_{i+1}}\}$:

$$\nu\tilde{\omega}.\theta_i^s\hat{\sigma}_{i+1}^0\sigma_L \approx_s \nu\tilde{\omega}.\theta_i^s\hat{\sigma}_{i+1}^0\sigma_R.$$

Now we consider $R_{i+1} = \mathsf{d}(\mathsf{p}(\mathsf{id}_{i+1}), \mathsf{dec}(\Pi_2(M_{i+1}), \mathsf{a}_3))$ and, according to this definition, we consider several cases:

- Either $i + 1 \in \{1, 2\}$, then, according to Lemma 9, $\mathsf{p}(\mathsf{id}_{i+1})$ is not deducible for $\theta_i^s\hat{\sigma}_i^0\sigma$ and so is R_{i+1} for $\sigma \in \{\sigma_L, \sigma_R\}$. Moreover, $R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma$ is neither a subterm of $\theta_i^s\hat{\sigma}_i^0\sigma$ for $\sigma \in \{\sigma_L, \sigma_R\}$, nor deducible from it, then, since d is a destructor in E, we can use Lemma 7 to conclude.

- Or $i + 1 \geq 3$. Using Lemma 19, we have $\Pi_2(M_2)\theta_i^s\hat{\sigma}_i^0\sigma =_E \mathsf{blind}(\mathsf{penc}(P_1, P_2, U), P_3)\theta_i^s\hat{\sigma}_i^0\sigma$ with free terms P_1, P_2, P_3 and free U or $U =_{AC} \mathsf{pk}(\mathsf{a}_\mathsf{p} + U')$ with free U' and $\mathsf{a}_\mathsf{p} \in \{\mathsf{a}_1, \mathsf{a}_3\}$ for $\sigma \in \{\sigma_L, \sigma_R\}$. We will consider two different subcases:

  - $U\theta_i^s\hat{\sigma}_i^0\sigma \neq_E \mathsf{pk}(\mathsf{a}_3)$ for $\sigma \in \{\sigma_L, \sigma_R\}$. we introduce the following claims:

    Claim 1. $\forall x \in \mathsf{dom}(\theta_i^s\hat{\sigma}_i^0\sigma), x\theta_i^s\hat{\sigma}_i^0\sigma \neq_E \mathsf{dec}(M, \mathsf{a}_3)$.

    Claim 2. *If* $\theta_i^s\hat{\sigma}_i^0\sigma \vdash R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma$ *then* $R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma \in \mathsf{St}(\theta_i^s\hat{\sigma}_i^0\sigma)$.

    Claim 3. $R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma \notin \mathsf{St}(\theta_i^s\hat{\sigma}_i^0\sigma)$.

    According to Claim 2 and 3, we have that $R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma$ is neither a subterm of $\theta_i^s\hat{\sigma}_i^0\sigma$ nor appearing in it nor deducible from it, for $\sigma \in \{\sigma_L, \sigma_R\}$. We also have that, in normal form, $R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma =_E \mathsf{d}(\mathsf{p}(\mathsf{id}_{i+1}), \mathsf{dec}(\mathsf{blind}(\mathsf{penc}(P_1, P_2, U), P_3)\theta_i^s\hat{\sigma}_i^0\sigma, \mathsf{a}_3))$. Then, since d is a destructor in E, we can use Lemma 7 with previous equivalence to conclude.

  - $U\theta_i^s\hat{\sigma}_i^0\sigma =_E \mathsf{pk}(\mathsf{a}_3)$ for one $\sigma \in \{\sigma_L, \sigma_R\}$ then, using that $\theta_i^s\hat{\sigma}_i^0\sigma_L \approx_s \theta_i^s\hat{\sigma}_i^0\sigma_R$, we have $U\theta_i^s\hat{\sigma}_i^0\sigma =_E \mathsf{g}_3\theta_i^s\hat{\sigma}_i^0\sigma$ for $\sigma \in \{\sigma_L, \sigma_R\}$. In that case, we have $R_{i+1}\theta_i^s\hat{\sigma}_i^0\sigma =_E \mathsf{d}(\mathsf{p}(\mathsf{id}_{i+1}), \mathsf{blind}(P_1, P_3))\theta_i^s\hat{\sigma}_i^0\sigma$ with free P_1 and P_3. Thus, we can use Lemma 3 with previous equivalence to conclude.

$\square$

Finally, we show that we can add the final results outputted by the Decryption device to the frame providing that the Auditor has endorsed it.

Lemma 11. *Let us consider* $M_1,\ldots,M_n$ *free terms s.t.* $\forall\,i \in [\![1,n]\!]$, $\phi_\mathsf{R}(\mathsf{idp}_{i+1}, M_{i+1})\theta_i\hat\sigma_i^0\sigma_\mathsf{L} = \mathsf{ok}$ *with* $\mathsf{fv}(M_{i+1}) \subseteq \mathsf{dom}(\theta_i) \cup \mathsf{dom}(\hat\sigma_i^0) \setminus \{x_b^i\}$. *We suppose that* $\forall\,i \in [\![1,n]\!]$, $\nu\tilde\omega.\theta_i\hat\sigma_i^0\sigma_\mathsf{L} \approx_s \nu\tilde\omega.\theta_i\hat\sigma_i^0\sigma_\mathsf{R}$. *We also consider* $U_1,\ldots,U_n$ *and* $W_1,\ldots,W_n$ *be free terms such that:*

- $\mathsf{fv}(U_{k+1}) \subseteq \mathsf{dom}(\theta_n) \cup \{\mathsf{dom}(\hat\sigma_n^0)\} \cup \{d_1,\ldots,d_k\}$ *and* $\mathsf{fv}(W_{k+1}) \subseteq \mathsf{dom}(\theta_n) \cup \{\mathsf{dom}(\hat\sigma_n^k)\}$,

- $\phi_\mathsf{A}(\mathsf{hash}(\langle U_1,\ldots,U_n\rangle), hb_r^1,\ldots, hb_r^n, W_1,\ldots, W_n)\theta_n\hat\sigma\sigma_\mathsf{L} = \mathsf{ok}$, *with the corresponding notation* $hb_r^i = \langle\mathsf{idp}_i, \mathsf{hash}(\Pi_1(M_i))\rangle$.

Then, we have, for $\sigma \in \{\sigma_\mathsf{L},\sigma_\mathsf{R}\}$, $\forall\,1 \leq i \leq n$:

$$U_i\theta_n\hat\sigma\sigma =_E \mathsf{e}_\mathsf{i}\theta_n\hat\sigma\sigma, \text{ for } i = 1,2.$$

Moreover, there exist free terms U_1', U_2' *and free term* U_3' *or* $U_3' = \mathsf{pk}(\mathsf{a}_\mathsf{k} + U)$ *with free term* U:

$$U_i\theta_n\hat\sigma\sigma =_E \mathsf{penc}(U_1', U_2', U_3')\theta_n\hat\sigma\sigma, \text{ for } i \in [\![3,n]\!].$$

Proof. Since $\nu\tilde\omega.\theta_n\hat\sigma_n^0\sigma_L \approx_s \nu\tilde\omega.\theta_n\hat\sigma_n^0\sigma_R$, then $\nu\tilde\omega.\theta_n\hat\sigma_n^n\sigma_L \approx_s \nu\tilde\omega.\theta_n\hat\sigma_n^n\sigma_R$ and according to the fact that ϕ_A is a public test, we also know that, for $\sigma \in \{\sigma_L,\sigma_R\}$:

$$\phi_\mathsf{A}(\mathsf{hash}(\langle U_1,\ldots,U_n\rangle), hb_r^1,\ldots, hb_r^n, W_1,\ldots, W_n)\theta_n\hat\sigma_n^n\sigma = \mathsf{ok}.$$

This implies that, according to the definition of ϕ_A, that:

$$\mathsf{hash}(\langle U_1,\ldots,U_n\rangle)\theta_n\hat\sigma_n^n\sigma =_E \mathsf{hash}(\langle\Pi_2(W_1),\ldots,\Pi_2(W_n)\rangle)\theta_n\hat\sigma_n^n\sigma$$
$$\mathsf{hash}(\Pi_1(M_i))\theta_n\hat\sigma_n^n\sigma =_E \mathsf{hash}(W_i)\theta_n\hat\sigma_n^n\sigma, \text{ for } i \in [\![1,n]\!]$$

From these two equations, we can deduce that $\forall i \in [\![1,n]\!]$, we have:

$$U_i\theta_n\hat\sigma_n^n\sigma =_E \Pi_2(\Pi_1(M_i))\theta_n\hat\sigma_n^n\sigma.$$

According to the construction of θ_i and $\hat\sigma_i^j$, we have that, $\forall\,i \in [\![0,n]\!]$, $\theta_i = \theta_n\,|_{\mathsf{dom}(\theta_i)}$ and $\hat\sigma_i^0 = \hat\sigma_n^0\,|_{\mathsf{dom}(\hat\sigma_i^0)}$. In particular, we also have that, $\forall i \in [\![1,n]\!]$, $\mathsf{fv}(M_i) \subseteq \mathsf{dom}(\theta_{i-1})$ and $\mathsf{fv}(M_i\theta_{i-1}) \subseteq \mathsf{dom}(\hat\sigma_{i-1}^0)$. Thus, we have:

$$\forall i \in [\![1,n]\!], \ M_i\theta_n\hat\sigma_n^0\sigma =_E M_i\theta_{i-1}\hat\sigma_{i-1}^0\sigma.$$

But, according to Lemma 19, we know that $\Pi_1(M_i)\theta_{i-1}\hat\sigma_{i-1}^0\sigma$ is an id_i-valid ballot. Thus, using Lemma 18 for $i \in [\![3,n]\!]$ we have that:

$$\Pi_2(\Pi_1(M_i))\theta_{i-1}\hat\sigma_{i-1}^0\sigma =_E \begin{cases} \mathsf{e}_\mathsf{i}\theta_{i-1}\hat\sigma_{i-1}^0\sigma \text{ for } i \in \{1,2\} \\ \mathsf{penc}(N,P,Q)\theta_{i-1}\hat\sigma_{i-1}^0\sigma, \text{ for } i \in [\![3,n]\!], \end{cases}$$

with N and P free terms and Q a free term or $Q = \mathsf{pk}(\mathsf{a}_\mathsf{k} + Q')$ with Q' a free term and such that $\mathsf{fv}(N) \cup \mathsf{fv}(P) \cup \mathsf{fv}(Q) \subseteq \mathsf{fv}(M_i)$. According to this and the fact that $\mathsf{fv}(M_i\theta_n) \subseteq \mathsf{dom}(\hat\sigma_n^0)$ and that, for $i \in [\![0,n]\!]$, $\hat\sigma_n^i = \hat\sigma_n^n\,|_{\mathsf{dom}(\hat\sigma_n^i)}$, we have that the previous result holds for $\hat\sigma_n^n$ instead of $\hat\sigma_n^0$ and we conclude:

$$U_i\theta_n\hat\sigma_n^n\sigma =_E \begin{cases} \mathsf{e}_\mathsf{i}\theta_n\hat\sigma_n^n\sigma \text{ for } i \in \{1,2\} \\ \mathsf{penc}(N,P,Q)\theta_n\hat\sigma_n^n\sigma, \text{ for } i \in [\![3,n]\!]. \end{cases}$$

$\square$

F Defining the Relation $\mathcal{R}$

In this section, we first define the different partial evolutions of processes that are needed to present the relation itself. Then, we describe the relation $\mathcal{R}$ and provide a proof for Proposition 2. From now and for the rest of the proofs, the randomnesses r_1 and r_2 (which are the honest voters randomnesses) have been replaced by t_1 and t_2 for notations issues.

F.1 Partial Evolutions

Let us recall some notations introduced in Section 3: $\sigma_L = \{^a/_{v_1}, ^b/_{v_2}\}$ and $\sigma_R = \{^b/_{v_1}, ^a/_{v_2}\}$, two substitutions that summarize voting choices made by honest voters during processes, and $\tilde{n}_b = a_1, a_3, id_1, id_2, t_1, t_2, id_R, c_{RV_1}, c_{RV_2}, c_{RA}, c_{DA}, c_{RD}$, a set of names representing restricted names in the process. These names are corresponding to secret keys, ids and channels.

Now, we can define the different partial evolutions of the multiple processes involved in the overall process P_n^b of the Norwegian protocol, as described in Section 3.2.6.

Definition 10. *We decompose the process of the (honest) voter using the following notations (we remind that $V_i(v_i) = V(c_i, c_{RV_i}, c_{out}, g_1, id_i, idp_R, v_i)$):*

$$
\begin{array}{llll}
V_1^i &=& \overline{c_{out}}\langle ball_i\rangle . V_2^i & \qquad V_5^i &=& c_i(x_b^i) . V_6^i \\
V_2^i &=& \overline{c_i}\langle ball_i\rangle . V_3^i & \qquad V_6^i &=& if\ \phi_V(id_i, idp_R, v_i, e_i, p_i, si_i, x_b^i, x_r^i)\ then\ V_7^i \\
V_3^i &=& \overline{c_{RV_i}}\langle ok\rangle . V_4^i & \qquad V_7^i &=& \overline{c_{out}}\langle ok\rangle . V_8^i \\
V_4^i &=& c_{RV_i}(x_r^i) . V_5^i & \qquad V_8^i &=& \overline{c_{RV_i}}\langle ok\rangle
\end{array}
$$

with $ball_i = \langle e_i, p_i, si_i\rangle$, $e_i = penc(v_i, t_i, g_1)$, $p_i = pfk_1(id_i, t_i, v_i, e_i)$, *and* $si_i = sign(\langle e_i, p_i\rangle, id_i)$.

Definition 11. $R_n(c_{BR}, c_{RA}, c_{RD}, c_{out}, a_3, id_R, c_{RV_1}, idp_1, p(id_1), \ldots, c_{RV_n}, idp_n, p(id_n))$, *the process of the Receipt generator, is decomposed using the following notations:*

$$
\begin{array}{llll}
R_1^{i,n} &=& c_{RV_i}(go_r^i) . R_2^{i,n} & \qquad R_7^{i,n} &=& c_{RV_i}(sy_i) . R_1^{i+1,n} \\
R_2^{i,n} &=& c_{BR}(x_i) . R_3^{i,n} & \qquad R_7^{n,n} &=& c_{RV_n}(sy_n) . R_8^{1,n} \\
R_3^{i,n} &=& if\ \phi_R(idp_i, x_i)\ then\ R_4^{i,n} & \qquad R_8^{i,n} &=& \overline{c_{RA}}\langle hbr_i\rangle . R_8^{i+1,n} \\
R_4^{i,n} &=& \overline{c_{BR}}\langle si_r^i\rangle . R_5^{i,n} & \qquad R_8^{n,n} &=& \overline{c_{RA}}\langle hbr_n\rangle . R_9 \\
R_5^{i,n} &=& \overline{c_{out}}\langle r_i\rangle . R_6^{i,n} & \qquad R_9 &=& \overline{c_{RD}}\langle ok\rangle \\
R_6^{i,n} &=& \overline{c_{RV_i}}\langle r_i\rangle . R_7^{i,n}
\end{array}
$$

with $r_i = d(p(id_i), dec(\Pi_2(x_i), a_3))$, $si_r^i = sign(hash(\Pi_1(x_i)), id_R)$ *and* $hbr_i = \langle idp_i, hash(\Pi_1(x_i))\rangle$.

Definition 12. *We decompose $D_n(c_{BD}, c_{RD}, c_{DA}, c_{out}, a_1)$, the process of the decryption, device using the following notations:*

$$
\begin{array}{llll}
D_1 &=& c_{RD}(go_d^1) . D_2^{1,n} & \qquad D_4 &=& c_{DA}(go_d^2) . D_5^{[\![1,n]\!]} \\
D_2^{i,n} &=& c_{BD}(d_i) . D_2^{i+1,n} & \qquad D_5^S &=& \underset{j\in S}{|}\ \overline{c_{out}}\langle dec_j\rangle\ for\ S \in 2^{[\![1,n]\!]} \\
D_2^{n,n} &=& c_{BD}(d_n) . D_3 & \qquad D_5^\emptyset &=& 0 \\
D_3 &=& \overline{c_{DA}}\langle ha_d\rangle . D_4
\end{array}
$$

with $dec_i = dec(d_i, a_1)$ *and* $ha_d = hash(\langle d_1, \ldots, d_n\rangle)$.

Definition 13. *We decompose $A_n(c_{BA}, c_{RA}, c_{DA})$, the process of the auditor, using the notations below:*

$$
\begin{array}{llll}
A_1^{i,n} &=& c_{RA}(hr_i) . A_1^{i+1,n} & \qquad A_3^{n,n} &=& c_{BA}(hb_n) . A_4 \\
A_1^{n,n} &=& c_{RA}(hr_n) . A_2^{1,n} & \qquad A_4 &=& if\ \phi_A(h_d, hr_1, \ldots, hr_n, hb_1, \ldots, hb_n)\ then\ A_5 \\
A_2 &=& c_{DA}(h_d) . A_3^{1,n} & \qquad A_5 &=& \overline{c_{DA}}\langle ok\rangle \\
A_3^{i,n} &=& c_{BA}(hb_i) . A_3^{i+1,n}
\end{array}
$$

We introduce some more notations used as shorthands in the relation $\mathcal{R}$.

Definition 14. *We define two sets of names $\tilde{m} = \tilde{n}_b, go_r^1, go_r^2, x_r^1, x_r^2, sy_1, sy_2$ and $\tilde{h} = \tilde{m}, hr_1, \ldots, hr_n$. We remind that $\Gamma_b = \{^{pk(a_k)}/_{g_k}\ |\ k = 1..3\}\ |\ \{^{vk(id_k)}/_{idp_k}, ^{s(id_k)}/_{s_k}\ |\ k = 1, 2\}\ |\ \{^{vk(id_R)}/_{idp_R}\}$ and we*

consider the following frames:

$$\Gamma_{i,j,k} = \Gamma_b \mid \{^{\mathsf{ok}}/_{go^p_r} \mid \mathsf{p} = 1..i\} \mid \{^{\mathsf{ball_p}}/_{ba_p}, {}^{\mathsf{ball_p}}/_{ba'_p} \mid \mathsf{p} = 1..\min(i,2)\} \mid$$
$$\{^{r_p}/_{x^p_r} \mid \mathsf{p} = 1..j\} \mid \{^{\mathsf{ok}}/_{conf_p} \mid \mathsf{p} = 1..\min(k,2)\} \mid \{^{\mathsf{ok}}/_{sy_p} \mid \mathsf{p} = 1..k\}$$
$$\Lambda_i = \Gamma_{n,n,n} \mid \{^{hbr_p}/_{hr_p} \mid \mathsf{p} = 1..i\}$$
$$\Lambda = \Lambda_n \mid \{^{\mathsf{ok}}/_{go^1_d}, {}^{\mathsf{ok}}/_{go^2_d}, {}^{ha_d}/_{h_d}\}$$
$$\Delta_S = \Lambda \mid \{^{dec_{\delta_S(i)}}/_{res_i} \mid i = 1..\#\overline{S}\}$$
$$\text{with } \delta_S : \overline{S} \mapsto [\![1, \#\overline{S}]\!] \text{ a permutation and } \overline{S} = [\![1,n]\!] \setminus S \text{ where } S \in 2^{[\![1,n]\!]}.$$
$$\Delta_\delta = \Lambda \mid \{^{dec_{\delta(1)}}/_{res_1}, \dots, {}^{dec_{\delta(n)}}/_{res_n}\} \text{ with } \delta \text{ a permutation of } [\![1,n]\!].$$

We also define the following contexts:

$$C_0 \left[\_\right] = \nu\tilde{n}_b.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Gamma_c\right]$$
$$C_1^{i,j,k} \left[\_\right] = \nu\tilde{m}_{i,j,k}.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Gamma_{i,j,k}\right]$$
$$\text{with } \tilde{m}_{i,j,k} = \tilde{n}_c, go^1_r, \dots, go^i_r, x^1_r, \dots, x^j_r, sy_1, \dots, sy_k$$
$$C_2^{i,j,k} \left[\_\right] = \nu\tilde{m}.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Gamma_{i,j,k}\right]$$
$$C_3^i \left[\_\right] = \nu\tilde{m}, hr_1, \dots, hr_i.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Lambda_i\right]$$
$$C_4 \left[\_\right] = \nu\tilde{h}, go^1_d.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Lambda_n \mid \{^{\mathsf{ok}}/_{go^1_d}\}\right]$$
$$C_5 \left[\_\right] = \nu\tilde{h}, go^1_d, h_d.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Lambda_n \mid \{^{\mathsf{ok}}/_{go^1_d}\} \mid \{^{ha_d}/_{h_d}\}\right]$$
$$C_6^S \left[\_\right] = \nu\tilde{h}, go^1_d, go^2_d, h_d.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Delta_S\right]$$
$$C_7^\delta \left[\_\right] = \nu\tilde{h}, go^1_d, go^2_d, h_d.(let\ \mathsf{a}_3 = \mathsf{a}_1 + \mathsf{a}_2\ in\ ).\left[\_ \mid \Delta_\delta\right].$$

Using these notation, we can notice that we have:

$$P_n^b \left[V(\mathsf{c}_1, \mathsf{c}_{\mathsf{RV}_1}, \mathsf{c}_{\mathsf{out}}, \mathsf{g}_1, \mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1) \mid V(\mathsf{c}_2, \mathsf{c}_{\mathsf{RV}_2}, \mathsf{c}_{\mathsf{out}}, \mathsf{g}_1, \mathsf{id}_2, \mathsf{idp}_\mathsf{R}, \mathsf{v}_2)\right]$$
$$= C_0 \left[V_1^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n}\right].$$

F.1.1 Description of the Relation

We can now define the relation $\mathcal{R}$, based on all the previous definitions.

Definition 15. *For $i \in [\![1,n]\!]$, $j \in [\![1,2]\!]$, we define M_i, N_j, U_i and W_i free terms such that:*

- $\mathsf{fv}(M_i) \subseteq \mathsf{dom}(C_1^{i,i-1,i-1})$, $\mathsf{fn}(M_i) \cap \mathsf{bn}(C_1^{i,i-1,i-1}) = \emptyset$,

- $\mathsf{fv}(N_j) \subseteq \mathsf{dom}(C_1^{j,j,j-1})$, $\mathsf{fn}(N_j) \cap \mathsf{bn}(C_1^{j,j,i-1}) = \emptyset$,

- $\mathsf{fv}(U_i) \subseteq \mathsf{dom}(C_4)$, $\mathsf{fn}(U_i) \cap \mathsf{bn}(C_4) = \emptyset$,

- $\mathsf{fv}(W_i) \subseteq \mathsf{dom}(C_5)$, $\mathsf{fn}(W_i) \cap \mathsf{bn}(C_5) = \emptyset$.

We define the following abbreviations:

$$\widetilde{M}_k^{i,j} = \{^{M_p}/_{x_p} \mid \mathsf{p} = 1..k\} \mid \{^{si^p_r}/_{sr_p} \mid \mathsf{p} = 1..i\} \mid \{^{r_p}/_{rec_p} \mid \mathsf{p} = 1..j\},$$
$$\widetilde{N}_k = \{^{N_i}/_{x^i_b} \mid i = 1..k\},$$
$$\widetilde{M} = \widetilde{M}_n^{n,n} \mid \widetilde{N}_2,$$
$$\widetilde{U}_i = \widetilde{M} \mid \{^{U_p}/_{d_p} \mid \mathsf{p} = 1..i\},$$
$$\widetilde{W}_i = \widetilde{U}_n \mid \{^{W_p}/_{hb_p} \mid \mathsf{p} = 1..i\}.$$

We consider the smallest relation $\mathcal{R}$ which is closed under structural equivalence and includes the following pairs of extended processes.

%*Initialization*

$$C_0 \left[V_1^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \right] \sigma_L \; \mathcal{R} \; C_0 \left[V_1^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \right] \sigma_R \tag{1}$$

%*Output of encrypted ballot to a public channel (Voter 1)*

$$C_1^{0,0,0} \left[V_2^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \mid \{ {}^{\mathsf{ball}_1} /_{ba_1} \} \right] \sigma_L$$
$$\mathcal{R} \; C_1^{0,0,0} \left[V_2^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \mid \{ {}^{\mathsf{ball}_1} /_{ba_1} \} \right] \sigma_R \tag{2}$$

%*Output of encrypted ballot to B (Voter 1)*

$$C_1^{0,0,0} \left[V_3^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \mid \{ {}^{\mathsf{ball}_1} /_{ba_1}, {}^{\mathsf{ball}_1} /_{ba_1'} \} \right] \sigma_L$$
$$\mathcal{R} \; C_1^{0,0,0} \left[V_3^1 \mid V_1^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \mid \{ {}^{\mathsf{ball}_1} /_{ba_1}, {}^{\mathsf{ball}_1} /_{ba_1'} \} \right] \sigma_R \tag{3}$$

%*Synchronisation with the R (Voter 1)*

$$C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_2^{1,n} \mid D_1 \mid A_1^{1,n} \right] \sigma_L \; \mathcal{R} \; C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_2^{1,n} \mid D_1 \mid A_1^{1,n} \right] \sigma_R \tag{4}$$

%*R receives ballot coming from B (Voter 1)*

$$C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_3^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0} \right] \sigma_L$$
$$\mathcal{R} \; C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_3^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0} \right] \sigma_R \tag{5}$$

%*R's check failed (Voter 1)*

$$C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0} \right] \sigma_L \; \mathcal{R} \; C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0} \right] \sigma_R \tag{6}$$

%*R's check succeed (Voter 1)*

$$C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_4^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0} \right] \sigma_L$$
$$\mathcal{R} \; C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_4^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0} \right] \sigma_R \tag{7}$$

%*R outputs a signature for B (Voter 1)*

$$C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_5^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,0} \right] \sigma_L$$
$$\mathcal{R} \; C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_5^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,0} \right] \sigma_R \tag{8}$$

%*R outputs a signature to a public channel (Voter 1)*

$$C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_6^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \right] \sigma_L$$
$$\mathcal{R} \; C_1^{1,0,0} \left[V_4^1 \mid V_1^2 \mid R_6^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \right] \sigma_R \tag{9}$$

%*Reception of R's receipt (Voter 1)*

$$C_1^{1,1,0} \left[V_5^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,0} \left[V_5^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \right] \sigma_R \tag{10}$$

%*Reception of B's confirmation (Voter 1)*

$$C_1^{1,1,0} \left[V_6^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,0} \left[V_6^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{11}$$

%*Voter's check failed (Voter 1)*

$$C_1^{1,1,0} \left[V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,0} \left[V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{12}$$

%*Voter's check succeed (Voter 1)*

$$C_1^{1,1,0} \left[V_7^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,0} \left[V_7^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{13}$$

%*Voter outputs public confirmation (Voter 1)*

$$C_1^{1,1,0} \left[V_8^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \mid \{^{\mathsf{ok}}/_{conf_1}\} \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,0} \left[V_8^1 \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \mid \{^{\mathsf{ok}}/_{conf_1}\} \right] \sigma_R \tag{14}$$

%*Synchronisation with R and End Voter (Voter 1)*

$$C_1^{1,1,1} \left[V_1^2 \mid R_1^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,1} \left[V_1^2 \mid R_1^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{15}$$

%*Output of the encrypted ballot to a public channel (Voter 2)*

$$C_1^{1,1,1} \left[V_2^2 \mid R_1^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \mid \{^{\mathsf{ball_2}}/_{ba_2}\} \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,1} \left[V_2^2 \mid R_1^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \mid \{^{\mathsf{ball_2}}/_{ba_2}\} \right] \sigma_R \tag{16}$$

%*Output of the encrypted ballot to B (Voter 2)*

$$C_1^{1,1,1} \left[V_3^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \mid \{^{\mathsf{ball_2}}/_{ba_2}, {}^{\mathsf{ball_2}}/_{ba_2'}\} \right] \sigma_L$$
$$\mathcal{R}\ C_1^{1,1,1} \left[V_3^2 \mid R_1^{1,n} \mid D_1 \mid A_1^{1,n} \mid \{^{\mathsf{ball_2}}/_{ba_2}, {}^{\mathsf{ball_2}}/_{ba_2'}\} \right] \sigma_R \tag{17}$$

%*Synchronisation with the R (Voter 2)*

$$C_1^{2,1,1} \left[V_4^2 \mid R_2^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,1,1} \left[V_4^2 \mid R_2^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{18}$$

%*R receives ballot coming from B (Voter 2)*

$$C_1^{2,1,1} \left[V_4^2 \mid R_3^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,1,1} \left[V_4^2 \mid R_3^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{19}$$

%*R's check failed (Voter 2)*

$$C_1^{2,1,1} \left[V_4^2 \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{1,1} \mid \widetilde{N}_1 \right] \sigma_L \ \mathcal{R} \ C_1^{2,1,1} \left[V_4^2 \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{20}$$

%*R's check succeed (Voter 2)*

$$C_1^{2,1,1} \left[V_4^2 \mid R_4^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{1,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,1,1} \left[V_4^2 \mid R_4^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{1,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{21}$$

%*R outputs a signature to B (Voter 2)*

$$C_1^{2,1,1} \left[V_4^2 \mid R_5^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,1} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,1,1} \left[V_4^2 \mid R_5^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,1} \mid \widetilde{N}_1 \right] \sigma_R \tag{22}$$

%*R outputs a signature to a public channel (Voter 2)*

$$C_1^{2,1,1} \left[V_4^2 \mid R_6^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,1,1} \left[V_4^2 \mid R_6^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_1 \right] \sigma_R \tag{23}$$

%*Reception of R's receipt (Voter 2)*

$$C_1^{2,2,1} \left[V_5^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_1 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,2,1} \left[V_5^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_1 \right] \sigma_R \tag{24}$$

%*Reception of B's confirmation (Voter 2)*

$$C_1^{2,2,1} \left[V_6^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,2,1} \left[V_6^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \right] \sigma_R \tag{25}$$

%*Voter's check failed (Voter 2)*

$$C_1^{2,2,1} \left[R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \right] \sigma_L \ \mathcal{R} \ C_1^{2,2,1} \left[R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \right] \sigma_R \tag{26}$$

%*Voter's check succeed (Voter 2)*

$$C_1^{2,2,1} \left[V_7^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,2,1} \left[V_7^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \right] \sigma_R \tag{27}$$

%*Voter outputs public confirmation (Voter 2)*

$$C_1^{2,2,1} \left[V_8^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \mid \{^{\mathsf{ok}}/_{conf_2}\} \right] \sigma_L$$
$$\mathcal{R} \ C_1^{2,2,1} \left[V_8^2 \mid R_7^{2,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_2^{2,2} \mid \widetilde{N}_2 \mid \{^{\mathsf{ok}}/_{conf_2}\} \right] \sigma_R \tag{28}$$

The next relations describe the evolution of the protocol during the submissions of the intruder's ballots (corrupted voters). For $i \in [\![2, n-1]\!]$:

%*Synchronisation with R (Voter $i-1$)*

$$C_2^{i,i,i} \left[R_1^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_i^{i,i} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i,i,i} \left[R_1^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_i^{i,i} \mid \widetilde{N}_2 \right] \sigma_R \tag{29}$$

%*Synchronisation with R (Voter i)*

$$C_2^{i+1,i,i} \left[R_2^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_i^{i,i} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i+1,i,i} \left[R_2^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_i^{i,i} \mid \widetilde{N}_2 \right] \sigma_R \tag{30}$$

%*R receives the ballot coming from the intruder (Voter i)*

$$C_2^{i+1,i,i} \left[R_3^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i,i} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i+1,i,i} \left[R_3^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i,i} \mid \widetilde{N}_2 \right] \sigma_R \tag{31}$$

%*R's check failed (Voter i)*

$$C_2^{i+1,i,i} \left[D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i,i} \mid \widetilde{N}_2 \right] \sigma_L \; \mathcal{R} \; C_2^{i+1,i,i} \left[D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i,i} \mid \widetilde{N}_2 \right] \sigma_R \tag{32}$$

%*R's check succed (Voter i)*

$$C_2^{i+1,i,i} \left[R_4^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i,i} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i+1,i,i} \left[R_4^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i,i} \mid \widetilde{N}_2 \right] \sigma_R \tag{33}$$

%*R outputs signature to B (Voter i)*

$$C_2^{i+1,i,i} \left[R_5^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i+1,i} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i+1,i,i} \left[R_5^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i+1,i} \mid \widetilde{N}_2 \right] \sigma_R \tag{34}$$

%*R outputs receipt publicly (Voter i)*

$$C_2^{i+1,i,i} \left[R_6^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i+1,i+1} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i+1,i,i} \left[R_6^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i+1,i+1} \mid \widetilde{N}_2 \right] \sigma_R \tag{35}$$

%*R outputs receipt to Voter i*

$$C_2^{i+1,i+1,i} \left[R_7^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i+1,i+1} \mid \widetilde{N}_2 \right] \sigma_L$$
$$\mathcal{R} \; C_2^{i+1,i,i} \left[R_7^{i+1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_{i+1}^{i+1,i+1} \mid \widetilde{N}_2 \right] \sigma_R \tag{36}$$

The next two relations reflect the end of the voting phase and the beginning of the counting phase where R sends its content (hashed) to the Auditor. For $i \in [\![0, n-1]\!]$:

$$C_3^i \left[R_8^{i+1,n} \mid D_1 \mid A_1^{i+1,n} \mid \widetilde{M} \right] \sigma_L \; \mathcal{R} \; C_3^i \left[R_8^{i+1,n} \mid D_1 \mid A_1^{i+1,n} \mid \widetilde{M} \right] \sigma_R \tag{37}$$

$$C_3^n \left[R_9 \mid D_1 \mid A_2 \mid \widetilde{M} \right] \sigma_L \; \mathcal{R} \; C_3^n \left[R_9 \mid D_1 \mid A_2 \mid \widetilde{M} \right] \sigma_R \tag{38}$$

The final relations show the counting phase with the different steps of the Decryption device which receives encrypted votes from (corrupted) B and waits for Auditor's approval before it outputs the shuffled

result. For $i \in [\![1, n]\!]$:

$$C_4 \left[D_2^{i,n} \mid A_2 \mid \widetilde{U}_{i-1} \right] \sigma_L \; \mathcal{R} \; C_4 \left[D_2^{1,n} \mid A_2 \mid \widetilde{U}_{i-1} \right] \sigma_R \tag{39}$$

$$C_4 \left[D_3 \mid A_2 \mid \widetilde{U}_n \right] \sigma_L \; \mathcal{R} \; C_4 \left[D_3 \mid A_2 \mid \widetilde{U}_n \right] \sigma_R \tag{40}$$

$$C_5 \left[D_4 \mid A_3^{i,n} \mid \widetilde{W}_{i-1} \right] \sigma_L \; \mathcal{R} \; C_5 \left[D_4 \mid A_3^{i,n} \mid \widetilde{W}_{i-1} \right] \sigma_R \tag{41}$$

$$C_5 \left[D_4 \mid A_4 \mid \widetilde{W}_n \right] \sigma_L \; \mathcal{R} \; C_5 \left[D_4 \mid A_4 \mid \widetilde{W}_n \right] \sigma_R \tag{42}$$

$$C_5 \left[D_4 \mid \widetilde{W}_n \right] \sigma_L \; \mathcal{R} \; C_5 \left[D_4 \mid \widetilde{W}_n \right] \sigma_R \tag{43}$$

$$C_5 \left[D_4 \mid A_5 \mid \widetilde{W}_n \right] \sigma_L \; \mathcal{R} \; C_5 \left[D_4 \mid A_5 \mid \widetilde{W}_n \right] \sigma_R \tag{44}$$

For $S \in 2^{[\![1,n]\!]} \setminus \{\emptyset\}$, ${}^tS = S[1 \mapsto 2, 2 \mapsto 1]$ and $\delta_{{}^tS} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_S$ (with $\circ$ the classic composition of functions):

$$C_6^S \left[D_5^S \mid \widetilde{W}_n \right] \sigma_L \; \mathcal{R} \; C_6^{{}^tS} \left[D_5^{{}^tS} \mid \widetilde{W}_n \right] \sigma_R \tag{45}$$

For δ a permutation of $[\![1, n]\!]$ and ${}^t\delta$ s.t. ${}^t\delta = [1 \mapsto 2, 2 \mapsto 1] \circ \delta$:

$$C_7^\delta \left[\widetilde{W}_n \right] \sigma_L \; \mathcal{R} \; C_7^{{}^t\delta} \left[\widetilde{W}_n \right] \sigma_R \tag{46}$$

F.2 Proof of Observational Equivalence, Assuming Static Equivalence

In this section, we show that the relation $\mathcal{R}$ described before captures all the possible evolutions of the initial processes (Proposition 2). We first need to guarantee that the tests done by the voters and the Receipt generator pass (or fail) on both processes. This does not follow immediately from static equivalence of the frames since the tests done by the voters and teh Receipt generator are not public tests.

First, let us recall some notations:

$$\theta_0 = \{{}^{\mathsf{vk}(\mathsf{id}_k)}/_{\mathsf{idp}_k}, {}^{\mathsf{s}(\mathsf{id}_k)}/_{\mathsf{s}_k} \mid \mathsf{k} = 1..n\} \mid \{{}^{\mathsf{vk}(\mathsf{id}_R)}/_{\mathsf{idp}_R}\} \mid \{{}^{\mathsf{pk}(\mathsf{a}_k)}/_{\mathsf{g}_k} \mid \mathsf{k} = 1..3\} \mid$$
$$\{{}^{\mathsf{penc}(\mathsf{v}_k,\mathsf{t}_k,\mathsf{g}_1)}/_{\mathsf{e}_k}, {}^{\mathsf{pfk}_1(\mathsf{id}_k,\mathsf{t}_k,\mathsf{v}_k,\mathsf{e}_k)}/_{\mathsf{p}_k}, {}^{\mathsf{sign}(\langle \mathsf{e}_k,\mathsf{p}_k \rangle,\mathsf{id}_k)}/_{\mathsf{si}_k} \mid \mathsf{k} = 1..2\},$$
$$\theta_k = \theta_{k-1} \mid \{{}^{\mathsf{sign}(\mathsf{hash}(\Pi_1(x_k)),\mathsf{id}_R)}/_{sr_k}, {}^{\mathsf{d}(\mathsf{p}(\mathsf{id}_k),\mathsf{dec}(\Pi_2(x_k),\mathsf{a}_3))}/_{rec_k}\},$$
$$\sigma_i^j = \{{}^{M_k}/_{x_k} \mid \mathsf{k} = 1..i\} \mid \{{}^{N_k}/_{x_b^k} \mid \mathsf{k} = 1..\min(i,2)\} \mid \{{}^{U_k}/_{d_k}, {}^{W_k}/_{hb_k} \mid \mathsf{k} = 1..j\}.$$

Lemma 20. *Let $i \in \{1, 2\}$ and let M and N be two free terms and let be $P = \mathsf{d}(\mathsf{p}(\mathsf{id}_i), \mathsf{dec}(\Pi_2(N), \mathsf{a}_3))$. We suppose that $\phi_R(\mathsf{idp}_i, N)\theta_{i-1}\sigma_{i-1}^0\sigma_L = \mathsf{ok}$ and $\phi_V(\mathsf{id}_i, \mathsf{idp}_R, \mathsf{v}_i, \mathsf{e}_i, \mathsf{p}_i, \mathsf{si}_i, M, P)\theta_i\sigma_i^0\sigma_L = \mathsf{ok}$.*
Then, we have $\phi_V(\mathsf{id}_i, \mathsf{idp}_R, \mathsf{v}_i, \mathsf{e}_i, \mathsf{p}_i, \mathsf{si}_i, M, P)\theta_i\sigma_i^0\sigma_R = \mathsf{ok}$.

Proof. Let $i \in \{1, 2\}$. Since $\phi_V(\mathsf{id}_i, \mathsf{idp}_R, \mathsf{v}_i, \mathsf{e}_i, \mathsf{p}_i, \mathsf{si}_i, M, P)\theta_i\sigma_i^0\sigma_L = \mathsf{ok}$, we have, in particular, that:

$$(\mathsf{checksign}(\mathsf{hash}(\langle \mathsf{idp}_i, \mathsf{e}_i, \mathsf{p}_i, \mathsf{si}_i \rangle), \mathsf{idp}_R, M) =_E \mathsf{ok})\theta_i\sigma_i^0\sigma_L.$$

Since this test is public (is uses no restricted names), and since $\theta_i\sigma_i^0\sigma_L \approx_s \theta_i\sigma_i^0\sigma_R$ (Proposition 1):

$$(\mathsf{checksign}(\mathsf{hash}(\langle \mathsf{idp}_i, \mathsf{e}_i, \mathsf{p}_i, \mathsf{si}_i \rangle), \mathsf{idp}_R, M) =_E \mathsf{ok})\theta_i\sigma_i^0\sigma_R.$$

We know that $(\phi_R(\mathsf{idp}_i, N) =_E \mathsf{ok})\theta_{i-1}\sigma_{i-1}^0\sigma_L$. Thus, according to Lemma 19, for $\sigma \in \{\sigma_L, \sigma_R\}$:

- $(\phi_R(\mathsf{idp}_i, N) =_E \mathsf{ok})\theta_{i-1}\sigma_{i-1}^0\sigma$,

- $(\Pi_2(N) =_E \mathsf{blind}(\mathsf{renc}(\mathsf{e}_i, Q_1), Q_2))\theta_{i-1}\sigma_{i-1}^0\sigma$ for free terms Q_1, Q_2.

Thus, we have, for $\sigma \in \{\sigma_L, \sigma_R\}$:

$$P\theta_i\sigma_i^0\sigma =_E \mathsf{d}(\mathsf{p}(\mathsf{id}_\mathsf{i}), \mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\mathsf{e}_\mathsf{i}, Q_1), Q_2), \mathsf{a}_3))\theta_i\sigma_i^0\sigma.$$

Using the fact that $\mathsf{d}(\mathsf{p}(\mathsf{id}_\mathsf{i}), \mathsf{blind}(\mathsf{v}_\mathsf{i}, \mathsf{s}(\mathsf{id}_\mathsf{i})))\theta_i\sigma_i^0\sigma_L = P\theta_i\sigma_i^0\sigma_L$, we must have:

$$\mathsf{blind}(\mathsf{v}_\mathsf{i}, \mathsf{s}(\mathsf{id}_\mathsf{i}))\theta_i\sigma_i^0\sigma_L = \mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\mathsf{e}_\mathsf{i}, Q_1), Q_2), \mathsf{a}_3)\theta_i\sigma_i^0\sigma_L,$$

which leads us to the fact that $(Q_1 =_E \mathsf{a}_2)\theta_i\sigma_i^0\sigma_L$ and $(Q_2 =_E \mathsf{s}_\mathsf{i})\theta_i\sigma_i^0\sigma_L$. Thus, since $\theta_i\sigma_i^0\sigma_L \approx_s \theta_i\sigma_i^0\sigma_R$, those equalities holds with $\theta_i\sigma_i^0\sigma_R$ and we can finally state that:

$$\mathsf{blind}(\mathsf{v}_\mathsf{i}, \mathsf{s}(\mathsf{id}_\mathsf{i}))\theta_i\sigma_i^0\sigma_R = \mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\mathsf{e}_\mathsf{i}, Q_1), Q_2), \mathsf{a}_3)\theta_i\sigma_i^0\sigma_R,$$

and we conclude.

$\square$

We are now ready to prove Proposition 2.

Proposition 2. *Let $\mathcal{R}$ be the relation defined in Definition 15 (Section F). Then, $\mathcal{R}$ is verifying properties* (2) *and* (3) *of Definition 4.*

Proof. Let us prove that $\mathcal{R}$ verifies properties (2) and (3) of Definition 4.

INTERNAL REDUCTIONS: We must show for all extended processes A and B, where $A\mathcal{R}B$, that if $A \to A'$ for some A', then $B \to^* B'$ with $A'\mathcal{R}B'$ for some B'. We observe that if $A\mathcal{R}B$ by (1), (2), (4), (6) to (8), (10), (12), (13), (15), (16), (18), (20) to (22), (24), (26), (27), (29), (30), (32) to (36), (39), (41), (43), (45) or (46), then there is no extended process A' such that $A \to A'$. We proceed by case analysis depending on which relation is satisfied:

(3) In that case, we have $A \equiv Z_0\left[V_3^1 \mid R_1^{1,n}\right]\sigma_L$ and $B \equiv Z_0\left[V_3^1 \mid R_1^{1,n}\right]\sigma_R$ with $Z_k[\_] = C_1^{k,0,0}\left[\_ \mid V_1^2 \mid D_1 \mid A_1^{1,n} \mid \{^{\mathsf{ball}_1}/_{b}\right.$

If $A \to A'$, it must be the case that $A \equiv Z_0\left[\overline{\mathsf{c}_{\mathsf{RV}_1}}\langle\mathsf{ok}\rangle \cdot V_4^1 \mid \mathsf{c}_{\mathsf{RV}_1}(go_r^1) \cdot R_2^{1,n}\right]\sigma_L$ and we have $A' \equiv Z_1\left[V_4^1 \mid R_2^{1,n}\right]\sigma_L$. Now, since we have that $B \equiv Z_0\left[\overline{\mathsf{c}_{\mathsf{RV}_1}}\langle\mathsf{ok}\rangle \cdot V_4^1 \mid \mathsf{c}_{\mathsf{RV}_1}(go_r^1) \cdot R_2^{1,n}\right]\sigma_R$, it follows that $B \to B'$ where $B' = Z_1\left[V_4^1 \mid R_2^{1,n}\right]\sigma_R$ and we derive $A'\mathcal{R}B'$ by relation (4) and the closure of $\mathcal{R}$ under structural equivalence.

(5) In that case, we have $A \equiv Z\left[R_3^{1,n}\right]\sigma_L$ and $B \equiv Z\left[R_3^{1,n}\right]\sigma_R$ with the context $Z[\_] = C_1^{1,0,0}\left[\_ \mid V_4^1 \mid V_1^2 \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{0,0}\right.$ and a term M_1 such that $\mathsf{fv}(M_1) \subseteq \mathsf{dom}(C_1^{1,0,0})$ and $\mathsf{fn}(M_1) \cap \mathsf{bn}(C_1^{1,0,0}) = \emptyset$. If $A \to A'$, then $A \equiv Z\left[\text{if } \phi_\mathsf{R}(\mathsf{idp}_1, x_1) \text{ then } R_4^{1,n}\right]\sigma_L$ and either $A' \equiv Z\left[R_4^{1,n}\right]\sigma_L$ or $A' \equiv Z[0]\sigma_L$. In the first case (resp. the second case), $\phi_\mathsf{R}(\mathsf{idp}_1, M_1)\theta_0\sigma_0^0\sigma_L$, which is equivalent to $\phi_\mathsf{R}(\mathsf{idp}_1, x_1)\sigma_L$ in the Z context, must be equivalent (resp. not equivalent) to ok in E. Then, using Lemma 19, we know that $\phi_\mathsf{R}(\mathsf{idp}_1, M_1)\theta_0\sigma_0^0\sigma_R$ (i.e. $\phi_\mathsf{R}(\mathsf{idp}_1, x_1)\sigma_R$ in the Z context) is also equivalent (resp. not equivalent) to ok in E. It follows from $B \equiv Z\left[\text{if } \phi_\mathsf{R}(\mathsf{idp}_1, x_1) \text{ then } R_4^{1,n}\right]\sigma_R$ that $B \to B'$ with $B' = Z\left[R_4^{1,n}\right]\sigma_R$ (resp. $B' = Z[0]\sigma_R$) which leads us to relation (7) (resp. (6)) and we conclude.

(11) In that case, we have $A \equiv Z\left[V_6^1\right]\sigma_L$ and $B \equiv Z\left[V_6^1\right]\sigma_R$ with the context $Z[\_] = C_1^{1,1,0}\left[\_ \mid V_1^2 \mid R_7^{1,n} \mid D_1 \mid A_1^{1,n} \mid \widetilde{M}_1^{1,1} \mid\right.$ with terms M_1, N_1 s.t. $\mathsf{fv}(M_1) \subseteq \mathsf{dom}(C_1^{1,0,0})$, $\mathsf{fv}(N_1) \subseteq \mathsf{dom}(C_1^{1,1,0})$, $\mathsf{fn}(M_1) \cap \mathsf{bn}(C_1^{1,0,0}) = \emptyset$ and $\mathsf{fn}(N_1)\cap\mathsf{bn}(C_1^{1,1,0}) = \emptyset$. If $A \to A'$, we have $A \equiv Z\left[\text{if } \phi_\mathsf{V}(\mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1, \mathsf{e}_1, \mathsf{p}_1, \mathsf{si}_1, x_b^1, x_r^1) \text{ then } V_7^1\right]\sigma_L$ and either $A' \equiv Z\left[V_7^1\right]\sigma_L$ or $A' \equiv Z[0]\sigma_L$. In the first (resp. the second) case, $\phi_\mathsf{V}(\mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1, \mathsf{e}_1, \mathsf{p}_1, \mathsf{si}_1, N_1, P)\theta_1\sigma_1^0\sigma_L$ with $P = \mathsf{d}(\mathsf{p}(\mathsf{id}_1), \mathsf{dec}(\Pi_2(M_1), \mathsf{a}_3))$, equivalent to $\phi_\mathsf{V}(\mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1, \mathsf{e}_1, \mathsf{p}_1, \mathsf{si}_1, x_b^1, x_r^1)\sigma_L$ in the Z context, must be equivalent (resp. not equivalent) to ok in E. Then, using Lemma 20, $\phi_\mathsf{V}(\mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1, \mathsf{e}_1, \mathsf{p}_1, \mathsf{si}_1, N_1, P)\theta_1\sigma_1^0\sigma_R$ (i.e. $\phi_\mathsf{V}(\mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1, \mathsf{e}_1, \mathsf{p}_1, \mathsf{si}_1, x_b^1, x_r^1)\sigma_R$ in the Z context) is also equivalent (resp. not equivalent) to ok in E. It follows from the following equivalence $B \equiv Z\left[\text{if } \phi_\mathsf{V}(\mathsf{id}_1, \mathsf{idp}_\mathsf{R}, \mathsf{v}_1, \mathsf{e}_1, \mathsf{p}_1, \mathsf{si}_1, x_b^1, x_r^1) \text{ then } V_7^1\right]\sigma_R$ that we have $B \to B'$ with $B' \equiv Z\left[V_7^1\right]\sigma_R$ (resp. $B' \equiv Z[0]\sigma_R$) which leads us to relation (13) (resp. (12)) and we conclude.

(42) In that case, we have $A \equiv Z\,[A_4]\,\sigma_L$ and $B \equiv Z\,[A_4]\,\sigma_R$ with the context $Z[\_] = C_5\left[\_\mid D_4\mid \widetilde{W_n}\right]$
and free terms $M_1, \ldots, M_n, N_1, N_2, U_1, \ldots, U_n$ and $W_1, \ldots, W_n$ such that $\mathsf{fv}(M_i) \subseteq \mathsf{dom}(C_1^{i+1,i,i})$, $\mathsf{fn}(M_i) \cap \mathsf{bn}(C_1^{i+1,i,i}) = \emptyset$ for $i \in [\![1, n]\!]$, $\mathsf{fv}(N_i) \subseteq \mathsf{dom}(C_1^{i,i,0})$, $\mathsf{fn}(N_i) \cap \mathsf{bn}(C_1^{i,i,0}) = \emptyset$ for $i \in \{1, 2\}$, $\mathsf{fv}(U_i) \subseteq \mathsf{dom}(C_4)$, $\mathsf{fn}(U_i) \cap \mathsf{bn}(C_4) = \emptyset$ for $i \in [\![1, n]\!]$ and $\mathsf{fv}(W_i) \subseteq \mathsf{dom}(C_5)$, $\mathsf{fn}(W_i) \cap \mathsf{bn}(C_5) = \emptyset$ for $i \in [\![1, n]\!]$. If $A \to A'$, we must have the following equivalence $A \equiv Z\,[\mathsf{if}\ \phi_\mathsf{A}(h_d, hr_1, \ldots, hr_n, hb_1, \ldots, hb_n)\ \mathsf{then}\ A_5]\,\sigma_L$ and either $A' \equiv Z\,[A_5]\,\sigma_L$ or $A' \equiv Z\,[0]\,\sigma_L$.
In the first case (resp. second case), we have $\phi_\mathsf{A}(\mathsf{hash}(\langle U_1, \ldots, U_n\rangle), hb_r^1, \ldots, hb_r^n, W_1, \ldots, W_n)\theta_n\sigma_n^n\sigma_L$ with $hb_r^i = \langle \mathsf{idp}_i, \mathsf{hash}(\Pi_1(M_i))\rangle$, which is equivalent, in the Z context, to $\phi_\mathsf{A}(h_d, hr_1, \ldots, hr_n, hb_1, \ldots, hb_n)\sigma_L$, must be equivalent (resp. not equivalent) to ok in E. Since this test is fully public and $\theta_n\sigma_n^n\sigma_L \approx_s \theta_n\sigma_n^n\sigma_R$, $\phi_\mathsf{A}(\mathsf{hash}(\langle U_1, \ldots, U_n\rangle), hb_r^1, \ldots, hb_r^n, W_1, \ldots, W_n)\theta_n\sigma_n^n\sigma_R$ (which is equivalent, in the Z context, to $\phi_\mathsf{A}(h_d, hr_1, \ldots, hr_n, hb_1, \ldots, hb_n)\sigma_R$) is also equivalent (resp. not equivalent) to ok in E. It follows from $B \equiv Z\,[\mathsf{if}\ \phi_\mathsf{A}(h_d, hr_1, \ldots, hr_n, hb_1, \ldots, hb_n)\ \mathsf{then}\ A_5]\,\sigma_R$ that we have $B \to B'$ with $B' \equiv Z\,[A_5]\,\sigma_R$ (resp. $B' \equiv Z\,[0]\,\sigma_R$) which leads us to relation (44) (resp. (43)) and we conclude.

The cases (9), (14), (17), (23), (28), (37), (38), (40) and (44) are similar to case (3), (19) is very similar to case (5) and, finally, (25) and (31) can be treated as case (11).

LABELLED REDUCTIONS: We must show for all extended processes A and B, where $A\mathcal{R}B$, that if $A \xrightarrow{\alpha} A'$ such that $\mathsf{fv}(\alpha) \subseteq \mathsf{dom}(A)$ and $\mathsf{bn}(\alpha) \cap \mathsf{fn}(B) = \emptyset$ for some A', then $B \to^* \xrightarrow{\alpha} \to^* B'$ with $A'\mathcal{R}B'$ for some B'. We observe that if $A\mathcal{R}B$ by (3), (5), (6), (9), (11), (12), (14), (17), (19), (20), (23), (25), (26), (28), (31), (32), (37), (38), (40), (42), (43), (44), or (46), then there is no extended process A' such that $A \xrightarrow{\alpha} A'$. We proceed by case analysis on the remaining cases:

(1) In that case, we have $A \equiv Z_0\left[V_1^1\right]\sigma_L$ and $B \equiv Z_0\left[V_1^1\right]\sigma_R$ with the context $Z_k[\_] = C_k^{0,0,0}\left[\_\mid V_2^1\mid R_1^{1,n}\mid D_1\mid A_1^{1,n}\right]$ (we consider $C_0 = C_0^{0,0,0}$). If $A \xrightarrow{\alpha} A'$ such that $\mathsf{fv}(\alpha) \subseteq \mathsf{dom}(A)$ and $\mathsf{bn}(\alpha) \cap \mathsf{fn}(B) = \emptyset$, then it must be the case that $A \equiv Z_0\left[\overline{\mathsf{c_{out}}}\langle\mathsf{ball}_1\rangle.V_2^1\right]\sigma_L$ and $A' \equiv Z_1\left[V_2^1\mid \{^{\mathsf{ball}_1}/_{ba_1}\}\right]\sigma_L$ where $\alpha = \nu ba_1.\overline{\mathsf{c_{out}}}\langle\mathsf{ball}_1\rangle$ and $ba_1 \notin \mathsf{dom}(Z_0)$. It follows from $B \equiv Z_0\left[\overline{\mathsf{c_{out}}}\langle\mathsf{ball}_1\rangle.V_2^1\right]\sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv Z_1\left[V_2^1\mid \{^{\mathsf{ball}_1}/_{ba_1}\}\right]\sigma_R$ and we derive $A'\mathcal{R}B'$ by relation (2) and the closure of $\mathcal{R}$ under structural equivalence.

(4) In that case, we have $A \equiv Z\left[R_2^{1,n}\right]\sigma_L$ and $B \equiv Z\left[R_2^{1,n}\right]\sigma_R$ with the context $Z[\_] = C_1^{1,0,0}\left[\_\mid V_4^1\mid V_1^2\mid D_1\mid A_1^{1,n}\right]$.
If $A \xrightarrow{\alpha} A'$ such that $\mathsf{fv}(\alpha) \subseteq \mathsf{dom}(A)$ and $\mathsf{bn}(\alpha) \cap \mathsf{fn}(B) = \emptyset$, then it must be the case that $A \equiv Z\left[\mathsf{c_{BR}}(x_1).R_3^{1,n}\right]\sigma_L$ and $A' \equiv Z\left[R_3^{1,n}\mid \widetilde{M_1^{0,0}}\right]\sigma_L$ where $\alpha = \mathsf{c_{BR}}(M_1)$ and a free term M_1 such that $\mathsf{fn}(M_1) \notin \mathsf{bn}(Z)$. It follows from $B \equiv Z\left[\mathsf{c_{BR}}(x_1).R_3^{1,n}\right]\sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv Z\left[R_3^{1,n}\mid \widetilde{M_1^{0,0}}\right]\sigma_R$ which leads us to relation (5) and we conclude.

(45) In that case, we have $A \equiv Z^S\left[D_5^S\right]\sigma_L$ and $B \equiv Z^{tS}\left[D_5^{tS}\right]\sigma_R$ with $Z^S[\_] = C_6^S\left[\_\mid \widetilde{W_n}\right]$, a set $S \in 2^{[\![1,n]\!]}\setminus\{\emptyset\}$, $^tS = S[1 \mapsto 2, 2 \mapsto 1]$, two permutations $\delta_S : \overline{S} \to [\![1, \#\overline{S}]\!]$ and $\delta_{tS} : \overline{{}^tS} \to [\![1, \#\overline{{}^tS}]\!]$ with $\overline{S} = [\![1, n]\!]\setminus S$ (note that $\#^tS = \#S$), and free terms $M_1, \ldots, M_n, N_1, N_2, U_1, \ldots, U_n$ and $W_1, \ldots, W_n$ such that $\mathsf{fv}(M_i) \subseteq \mathsf{dom}(C_1^{i+1,i,i})$, $\mathsf{fn}(M_i) \cap \mathsf{bn}(C_1^{i+1,i,i}) = \emptyset$ for $i \in [\![1, n]\!]$, $\mathsf{fv}(N_i) \subseteq \mathsf{dom}(C_1^{i,i,0})$, $\mathsf{fn}(N_i) \cap \mathsf{bn}(C_1^{i,i,0}) = \emptyset$ for $i \in \{1, 2\}$, $\mathsf{fv}(U_i) \subseteq \mathsf{dom}(C_4)$, $\mathsf{fn}(U_i) \cap \mathsf{bn}(C_4) = \emptyset$ for $i \in [\![1, n]\!]$ and $\mathsf{fv}(W_i) \subseteq \mathsf{dom}(C_5)$, $\mathsf{fn}(W_i) \cap \mathsf{bn}(C_5) = \emptyset$ for $i \in [\![1, n]\!]$. If $A \xrightarrow{\alpha} A'$ such that $\mathsf{fv}(\alpha) \subseteq \mathsf{dom}(A)$ and $\mathsf{bn}(\alpha) \cap \mathsf{fn}(B) = \emptyset$ then it must be the case that $A \equiv Z^S\left[\overline{\mathsf{c_{out}}}\langle dec_i\rangle.D_5^{S\setminus\{i\}}\right]\sigma_L$ with $i \in S$ and $\alpha = \nu res_j.\overline{\mathsf{c_{out}}}\langle res_j\rangle$ and $res_j \notin \mathsf{dom}(Z^S)$ with $j = n + 1 - \#S$. Then, we consider two cases:

- Either $i \in [\![3, n]\!]$. Let $S' = S\setminus\{i\}$. We consider two cases:

  - If $S' \neq \emptyset$ (i.e. $j < n$), we have $\overline{S'} = \overline{S}\cup\{i\}$. Thus, $A' \equiv Z^{S'}\left[D_5^{S'}\right]\sigma_L$ with $\delta_{S'} : \overline{S'} \to [\![1, \#\overline{S'}]\!]$ and $\delta_{S'} = \delta_S \circ [j \mapsto i]$. Since $i \in [\![3, n]\!]$, we also have that

${}^t S' = {}^t S \setminus \{i\}$, thus, it follows from $B \equiv Z^{{}^t S} \left[\overline{\mathsf{c_{out}}} \langle dec_i \rangle . D_5^{{}^t S \setminus \{i\}} \right] \sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv Z^{{}^t S'} \left[D_5^{{}^t S'} \right] \sigma_R$ with $\delta_{{}^t S'} = \delta_{{}^t S} \circ [j \mapsto i]$. Since we have $\delta_{{}^t S} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_S$ and since $\delta_{{}^t S'}(j) = \delta_{S'}(j) = i$, we still have that $\delta_{{}^t S'} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_{S'}$ which leads us back to relation (45) and we conclude.

- $S' = \emptyset$ (i.e. $j = n$), we have $\overline{S'} = [\![1, n]\!]$. Thus $A' \equiv C_7^\delta \left[\widetilde{W_n} \right] \sigma_L$ with $\delta = \delta_S \circ [n \mapsto i]$. it follows from $B \equiv Z^{{}^t S} \left[\overline{\mathsf{c_{out}}} \langle dec_i \rangle \right] \sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv C_7^{{}^t \delta} \left[\widetilde{W_n} \right] \sigma_R$ with ${}^t \delta = \delta_{{}^t S} \circ [n \mapsto i]$. Since we have $\delta_{{}^t S} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_S$ and since $\delta_{{}^t S'}(n) = \delta_{S'}(n) = j$, we still have that $\delta_{{}^t S'} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_{S'}$ which leads us to relation (46) and we conclude.

- Or $i \in \{1, 2\}$. W.l.o.g., suppose that $i = 1$ and let $S' = S \setminus \{1\}$. We consider two cases:

  - If $S' \neq \emptyset$ (i.e. $j < n$), we have $\overline{S'} = \overline{S} \cup \{1\}$. Thus, $A' \equiv Z^{S'} \left[D_5^{S'} \right] \sigma_L$ with $\delta_{S'} : \overline{S'} \rightarrow [\![1, \#\overline{S'}]\!]$ and $\delta_{S'} = \delta_S \circ [j \mapsto 1]$. Since $1 \in S$, we know, by definition of ${}^t S$, that $2 \in {}^t S$. Thus, we have that ${}^t S' = {}^t S \setminus \{2\}$, thus, it follows from $B \equiv Z^{{}^t S} \left[\overline{\mathsf{c_{out}}} \langle dec_2 \rangle . D_5^{{}^t S \setminus \{2\}} \right] \sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv Z^{{}^t S'} \left[D_5^{{}^t S'} \right] \sigma_R$ with $\delta_{{}^t S'} = \delta_{{}^t S} \circ [j \mapsto 2]$. Since we have $\delta_{{}^t S} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_S$ and since $\delta_{{}^t S'}(j) = 2 = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_{S'}(j)$, we still have that $\delta_{{}^t S'} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_{S'}$ which leads us back to relation (45) and we conclude.

  - $S' = \emptyset$ (i.e. $j = n$), we have $\overline{S'} = [\![1, n]\!]$. Thus $A' \equiv C_7^\delta \left[\widetilde{W_n} \right] \sigma_L$ with $\delta = \delta_S \circ [n \mapsto 1]$. it follows from $B \equiv Z^{{}^t S} \left[\overline{\mathsf{c_{out}}} \langle dec_2 \rangle \right] \sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv C_7^{{}^t \delta} \left[\widetilde{W_n} \right] \sigma_R$ with ${}^t \delta = \delta_{{}^t S} \circ [n \mapsto 2]$. Since we have $\delta_{{}^t S} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_S$ and since $\delta_{{}^t S'}(n) = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_{S'}(n)$, we still have that $\delta_{{}^t S'} = [1 \mapsto 2, 2 \mapsto 1] \circ \delta_{S'}$ which leads us to relation (46) and we conclude.

The cases (2), (7), (8), (13), (15), (16), (21), (22), (27), (33), (34) and (35) are similar to case (1). Cases (10), (18), (24), (29), (30), (36), (39) and (41) are very similar to (2).

<div align="right">□</div>