Computational Soundness of Formal Indistinguishability and Static Equivalence

Gergei Bana[*], Payman Mohassel, and Till Stegers

Department of Computer Science
University of California at Davis, USA
gebana@cs.upenn.edu mohassel@cs.ucdavis.edu stegers@cs.ucdavis.edu

Abstract. In the investigation of the relationship between the formal and the computational view of cryptography, a recent approach, first proposed in [10], uses static equivalence from cryptographic pi calculi as a notion of formal indistinguishability. Previous work [10, 1] has shown that this yields the soundness of natural interpretations of some interesting equational theories, such as certain cryptographic operations and a theory of XOR. In this paper however, we argue that static equivalence is too coarse to allow sound interpretations of many natural and useful equational theories. We illustrate this with several explicit examples in which static equivalence fails to work. To fix the problem, we propose a notion of formal indistinguishability that is more flexible than static equivalence. We provide a general framework along with general theorems, and then discuss how this new notion works for the explicit examples where static equivalence fails to ensure soundness.

1 Introduction

In the past few years, significant effort has been made to link formal and computational methods of cryptography. These directions had largely been developing independently; the first based on the seminal work of Dolev and Yao [15], and the second growing out of the work of Goldwasser and Micali [16]. While the computational method gives a more realistic and detailed description of an actual protocol, using probability theory and taking limited computational power into account, security proofs in this model are done by hand and are often notoriously hard to verify.

The formal method is a high-level treatment, amenable to automatization, but its reliability is sometimes questionable; namely, a protocol that is formally secure may not be so computationally, and, therefore, may be insecure in reality. It is therefore important how to translate one model into the other, and to characterize which security proofs in the simpler formal framework carry over to the computational setting.

The first paper to address this question was that of Abadi and Rogaway [4], which considered only passive adversaries. In their approach, the notion of

[*] Supported by a Packard Fellowship.

security is formalized via equivalence relations on the formal and on the computational side that specify which messages look indistinguishable to an adversary of the corresponding view. Fixing an encryption scheme for the computational implementation of the formal operations, a natural *interpretation* assigns a computational object – an ensemble of probability distributions over bit strings – to each formal expression. The question then arises: Under which circumstances is the equivalence relation preserved by the interpretation? If the formal equivalence of any two expressions implies the computational equivalence of their interpretations, then we say the model is *sound*. This is the mathematical equivalent of saying that security in the formal model implies security in the computational model. Conversely, the model is *complete* if the computational equivalence of the interpretations of any two formal expressions implies that these expressions are equivalent. Completeness of a model indicates that the formal equivalence notion in question is not too fine, and helps in finding attacks: if completeness holds, the existence of a formal attack implies the existence of a computational one.

Abadi and Rogaway proved soundness for their language if the encryption scheme used in the interpretation is what they call type-0 secure (basically, it hides everything about the plaintext). A number of other papers followed, proving completeness as well [20, 6], generalizing for weaker, more realistic encryptions schemes [6], considering purely probabilistic encryptions [17, 6], including limited models for active adversaries [19], and addressing the issue of forbidding key-cycles [5]. Other approaches including active adversaries are considered by Backes et al. and Canetti in their *reactive simulatability* [8, 7] and *universal composability* [11, 12] frameworks, respectively. Using probabilistic polynomial-time semantics without explicit probabilistic reasoning in [14] is also notable.

1.1 Previous Work

Our paper addresses issues when the equivalence relation on the formal side is *static equivalence* (from cryptographic pi calculi [3, 2]), induced by an *equational theory*. Equational theories model algebraic axioms in the formal world, such as axioms for groups, rings, XOR, etc. Once an equational theory is fixed, which means setting certain formal terms equal, static equivalence is uniquely determined. Roughly speaking, two n-tuples of formal terms are statically inequivalent, i.e. formally distinguishable, if an adversary is able to come up with two formal computations that, on one of the tuples yield two results that are identical according to the equational theory but yield different results on the other tuple. Baudet et al. [10] use this equivalence notion on the formal side, proving soundness of a theory of exclusive or as well as of certain symmetric encryptions that are deterministic and length-preserving. Abadi et al. [1] employ this framework to analyze a principled formal account of guessing attacks.

A shorter version of this paper [9], excluding proofs, was presented at ASIAN 2006.

1.2 Our Contributions

In this paper, we show that even though static equivalence works well to obtain soundness results for the cases analyzed in [10, 1], it does not work well in other important cases, and a more flexible notion is needed. For a brief exposition of why this is so, consider the Decisional Diffie-Hellman assumption. As Baudet et al. describe in [10], in an equational theory modeling group exponentiation without including logarithm, the 4-tuples (g, g^a, g^b, g^{ab}) and (g, g^a, g^b, g^c) are statically equivalent. Therefore, if the interpretation of the theory in a certain computational group scheme is sound, then this scheme satisfies the DDH assumption. However, formally much more is equivalent. For example, (g, g^a, g^b, ab) and (g, g^a, g^b, c) are also equivalent, and so on; an infinitude of statements *not necessarily implied* by the DDH assumption would be satisfied. There is no reason to think that such a computational group scheme exists at all. Moreover, the analysis often goes in the other direction: not a given formal model has to be interpreted in a sound manner, but for a given computational model we are looking for a formal theory that is simplifying, yet sound. A computational scheme that satisfies the DDH assumption may not satisfy the condition above (not to mention the infinitely many more that follow from static equivalence), so static equivalence cannot be used with such a group scheme to achieve soundness. Of course, if we *know* that the interpretations of two formal n-tuples are computationally distinguishable, then we may be able to incorporate the distinguisher into the formal theory, forcing those two n-tuples to be formally inequivalent. However, in many cases, we do not know whether the interpretations are inequivalent, so we have no explicit distinguishers. In such a case, to play it safe, it is better to assume that they are distinguishable, and that is how the formal theory should be constructed.

We argue that an equivalence relation finer than static equivalence is necessary to fit a number of interesting cases for which static equivalence is not suitable. We will call this type of equivalence relation a *formal indistinguishability relation* (FIR). We require four properties from any FIR, and through these properties an initial set of relations will generate a FIR. Each pair that is statically inequivalent is also inequivalent with respect to a formal indistinguishability relation. Moreover, static equivalence is one instance of a FIR. In order to test soundness with respect to a computational interpretation, it is enough to check soundness on a set of relations that generate the FIR in question. If soundness holds on the generating set of relations, then soundness holds in total.

Besides introducing the above equivalence notion, we also make some other improvements in the theory. Baudet et al. require the interpretations to be such that if a distribution is sampled twice, the probability of collision is negligible. We will not assume this because it would exclude the formal representation of interesting functions such as the least significant bit. We also use ordered sorts, allowing names to have multiple sorts.

After introducing the basic framework and proving some general propositions about FIRs, we discuss three examples. The first is the above-mentioned DDH assumption: We discuss how to introduce a FIR such that soundness is *equivalent* to the DDH assumption. Our second example considers the case of key-cycles and Laud's solution to them [18]. Laud proposed that if we do not want to exclude key-cycles from our theory and we do not want to assume that the encryption scheme is stronger than the usual assumptions (CPA, CCA-2, etc.), then we can simply assume that the formal adversary can decrypt all ciphertexts that were encrypted with keys that are in a key-cycle. We show how this assumption corresponds to a formal indistinguishability relation. Finally, the third example describes an embedding of Boolean propositional logic which fails to be sound with respect to static equivalence because two formal terms that are computationally distinguishable turn out to be statically equivalent.

We would like to thank Jonathan Herzog, Phillip Rogaway, and Andre Scedrov for valuable discussions on the topic.

2 Formal Model

2.1 Signatures, Terms, and Frames

A *signature* is a pair $(\mathfrak{S}, \mathcal{F})$, with $\mathfrak{S} = (\mathcal{S}, \mathcal{S}', \leq_{\mathcal{S}})$, $\mathcal{S}$ being a countably infinite set of *sorts* with partial order $\leq_{\mathcal{S}}$, $\mathcal{S}' \subseteq \mathcal{S}$, and $\mathcal{F}$ a finite set of *function symbols*. We use the notation $s, s_1, s_2, \ldots$ for sorts, and $f, f_1, f_2, \ldots$ for symbols. We assume that every $f \in \mathcal{F}$ has a unique *arity* $s_1 \times \cdots \times s_k \rightarrow s$ for some $s_1, \ldots, s_k, s \in \mathcal{S}$. If $k = 0$, then f is a *constant*, and we denote this as $f : s$.

Furthermore, let $\mathcal{X}$, $\mathcal{N}$ be countably infinite sets such that $\mathcal{S}$, $\mathcal{F}$, $\mathcal{X}$, $\mathcal{N}$ are pairwise disjoint. The elements of $\mathcal{X}$ are called *variables*, the elements of $\mathcal{N}$ *names*. We assume that both names and variables are sorted, that is, to each name or variable u, a subset $\mathcal{S}_u$ is assigned; we write $u : s$ and say u is of sort s whenever $s \in \mathcal{S}_u$. We require that $u : s_1$ and $s_1 \leq_{\mathcal{S}} s_2$ implies $u : s_2$, and that the set $\mathcal{S}_u$ has a minimum, which we denote by $\mathfrak{s}(u)$. For any subset U of the set of names or of the set of variables, let $[U]_s = \{u \in U \mid \mathfrak{s}(u) = s\}$. Finally, we require that for each sort s, $[\mathcal{X}]_s$ is infinite, and $[\mathcal{N}]_s$ is infinite whenever $s \in \mathcal{S}'$ and empty whenever $s \notin \mathcal{S}'$. A *renaming* is a bijection $\tau : \mathcal{N} \rightarrow \mathcal{N}$ such that $\mathfrak{s}(a) = \mathfrak{s}(\tau(a))$ for each name a. The terms of our language are sorted by elements of $\mathcal{S}$. As usual, if a term T has sort s, we write $T\colon s$. Terms of sort s are defined as follows:

$$T : s \ ::= \ x : s \mid a : s \mid f(T_1, \ldots, T_k)$$

where x is a variable, a is a name, and f is a function symbol of arity $s_1 \times \cdots \times s_k \rightarrow s'$ for some $s_1, \ldots, s_k \in \mathcal{S}$, $s' \leq_{\mathcal{S}} s$, and each term T_i is of sort s_i for $i = 1, \ldots, k$. The set of all terms will be denoted by $\mathcal{T}$. For a term T, we use $\mathrm{var}(T)$ for the set of variables occurring in T, and $\mathrm{names}(T)$ for the set of names occurring in T. A term T is said to be *closed* if $\mathrm{var}(T) = \emptyset$.

Let $x_1, \ldots, x_n$ be distinct variables, and let $T_1, \ldots, T_n$ be terms so that $\mathfrak{s}(T_i) \leq_{\mathcal{S}} \mathfrak{s}(x_i)$. A well-sorted *substitution* σ is written as $\sigma = \{x_1 = T_1, \ldots, x_n = T_n\}$. Since in this paper we will only have well-sorted substitutions, we will omit the term "well-sorted". The image of T under the substitution $\sigma = \{x_i = T_i\}_{i=1}^{n}$

is written as $T\sigma$, and is obtained by replacing every occurrence of x_i in T by T_i for each x_i. If all T_i are closed, σ is said to be *closed*; the *domain* of σ is $\mathrm{dom}(\sigma) = \{x_1, \ldots, x_n\}$ and the set of variables of σ is $\mathrm{var}(\sigma) = \bigcup_{i=1}^{n} \mathrm{var}(T_i)$. Similarly, we write $\mathrm{names}(\sigma)$ for the union of all sets $\mathrm{names}(T_i)$. As examples, we refer to the first paragraph of Subsection 5.1 or the second paragraph of 5.2.

Now we can define how to postulate axioms. In short, an *equational theory* is an equivalence relation on terms that is stable under (well-sorted) substitution of terms for variables, application of contexts, and renaming.

Definition 1. *An* equational theory *for a given signature is an equivalence relation $E \subseteq \mathcal{T} \times \mathcal{T}$ (written $=_E$ in infix notation) on the set of terms such that (i) $T =_E T'$ implies $T\sigma =_E T'\sigma$ for every substitution σ; (ii) $T_1 =_E T_2$ implies $T\{x = T_1\} =_E T\{x = T_2\}$ for every term T and every variable x with $\mathfrak{s}(x) \geq_{\mathcal{S}} \mathfrak{s}(T_i)$; (iii) $T_1 =_E T_2$ implies $\tau(T_1) =_E \tau(T_2)$ for every renaming τ.*

If R is a relation on $\mathcal{T}$, then the intersection $\langle R \rangle$ of all equational theories containing R is the smallest equational theory containing R. We say $\langle R \rangle$ is the equational theory *generated* by R. For examples, we refer to the second paragraph of Subsection 5.1 and the third paragraph of Subsection 5.2.

A *frame* φ is an expression $\nu\widetilde{a}.\sigma$, where σ is a substitution and $\widetilde{a} = \mathrm{names}(\sigma)$. Since $\widetilde{a}$ is uniquely determined by its *underlying substitution* σ, we may sometimes only write the substitution for a frame to save space. We say that φ is *closed* if σ is closed. The set of all frames (with respect to an understood signature) is denoted by $\mathfrak{F}$, the set of closed frames are denoted by $\mathfrak{F}_\mathrm{c}$.

If E is an equational theory and $\varphi = \nu\widetilde{a}.\sigma$ is a frame, we say that a term T is *deducible from* φ with respect to E, written $\varphi \vdash_E T$, if there is a term M with $\mathrm{var}(M) \subseteq \mathrm{dom}(\varphi)$ and $\mathrm{names}(M) \cap (\mathrm{names}(\varphi) \cup \mathrm{names}(T)) = \emptyset$ such that $M\varphi =_E T$, where $M\varphi = M\sigma$.

Suppose that for closed frames φ_1, φ_2 with $\mathrm{dom}(\varphi_1) = \mathrm{dom}(\varphi_2)$, there are two terms M, N sharing no names with φ_1 and φ_2, $\mathrm{var}(M) \cup \mathrm{var}(N) \subseteq \mathrm{var}(\varphi_i)$, such that $M\varphi_1 =_E N\varphi_1$, but $M\varphi_2 \neq_E N\varphi_2$. Intuitively, this means that carrying out two computations – permitted by the model and determined by M and N – on the inputs provided by φ_1, we get identical results, whereas carrying out the same computations on the input provided by φ_2 produces distinct results. If the distinction of two closed frames is not possible this way, then we say that these two frames are *statically equivalent.*

Definition 2. *Two closed frames φ_1, φ_2 of the same domain are* statically equivalent *with respect to an equational theory E, written $\varphi_1 \approx_E \varphi_2$, if for all terms M, N with $\mathrm{var}(M) \cup \mathrm{var}(N) \subseteq \mathrm{var}(\varphi_i)$ and using no names occurring in φ_1 or φ_2, we have $M\varphi_1 =_E N\varphi_1 \iff M\varphi_2 =_E N\varphi_2$. Let $\tilde{E}$ denote static equivalence as a subset of $\mathfrak{F}_\mathrm{c} \times \mathfrak{F}_\mathrm{c}$.*

2.2 Formal Indistinguishability

For a frame $\varphi = \nu \cup_{i=1}^{n} \mathrm{names}(T_i).\{x_i = T_i\}_{i=1}^{n}$, if φ' is another frame, let $\varphi\varphi'$ denote the frame $\nu \cup_{i=1}^{n} \mathrm{names}(T_i) \cup \mathrm{names}(\varphi').\{x_i = T_i\varphi'\}_{i=1}^{n}$. For frames

$\varphi_1, \ldots, \varphi_n$ with disjoint domains, let $\{\varphi_1|\varphi_2|\ldots|\varphi_n\}$ be the frame corresponding to the combination of all substitutions of $\varphi_1, \ldots, \varphi_n$.

Definition 3. *A* formal indistinguishability relation *with respect to an equational theory E is an equivalence relation $\cong$ on the set of closed frames such that*

(i) $\varphi_1 \cong \varphi_2$ *only if* $\mathrm{dom}(\varphi_1) = \mathrm{dom}(\varphi_2)$*;*
(ii) for any frame φ, if φ_1 and φ_2 are closed frames such that $\mathrm{var}(\varphi) \subseteq \mathrm{dom}(\varphi_\mathrm{i})$, $\mathrm{names}(\varphi) \cap \mathrm{names}(\varphi_i) = \emptyset$ *and $\varphi_1 \cong \varphi_2$ then $\varphi\varphi_1 \cong \varphi\varphi_2$;*
(iii) for any two frames $\varphi' = \{x_i = T'_i\}_{i=1}^n$ and $\varphi'' = \{x_i = T''_i\}_{i=1}^n$, if $T'_i =_E T''_i$ for all i, then $\varphi' \cong \varphi''$; moreover, $\varphi' \not\approx_E \varphi''$ implies $\varphi' \ncong \varphi''$;
(iv) for any renaming τ, $\tau(\varphi) \cong \varphi$.

Remark 1. Corresponding sections of equivalent frames are equivalent. That is, for example, if $\varphi_1 = \{x_i = T_i\}_{i=1}^4 \cong \varphi_2 = \{x_i = T'_i\}_{i=1}^4$, then $\{x_2 = T_2, x_4 = T_4\} \cong \{x_2 = T'_2, x_4 = T'_4\}$. This follows from (ii) by setting $\varphi = \nu\emptyset.\{x_2 = x_2, x_4 = x_4\}$.

If φ_1, φ_2, φ'_1, φ'_2 are frames such that $\mathrm{dom}(\varphi_1) \cap \mathrm{dom}(\varphi_2) = \emptyset$, $\mathrm{dom}(\varphi'_1) \cap \mathrm{dom}(\varphi'_2) = \emptyset$, $\mathrm{names}(\varphi_1) \cap \mathrm{names}(\varphi_2) = \emptyset$, $\mathrm{names}(\varphi'_1) \cap \mathrm{names}(\varphi'_2) = \emptyset$, and $\varphi_i \cong \varphi'_i$, then $\{\varphi_1|\varphi_2\} \cong \{\varphi'_1|\varphi'_2\}$. The reason is the following. Choose a renaming τ such that $\tau(\varphi_1) = \varphi_1$, $\tau(\varphi'_1) = \varphi'_1$, $\tau(\varphi'_2) = \varphi'_2$, and $\mathrm{names}(\tau(\varphi_2)) \cap \mathrm{names}(\varphi_1) = \mathrm{names}(\tau(\varphi_2)) \cap \mathrm{names}(\varphi'_1) = \emptyset$. This can be done because we assumed that there are infinitely many names of each sort. Using (iv), we see that $\{\varphi_1|\varphi_2\} \cong \tau(\{\varphi_1|\varphi_2\}) = \{\varphi_1|\tau(\varphi_2)\}$. If $\mathrm{dom}(\varphi_1) = \mathrm{dom}(\varphi'_1) = \{x_1, \ldots, x_k\}$, then let $\psi = \{x_1 = x_1, \ldots, x_k = x_k|\tau(\varphi_2)\}$. Using (ii), it follows that $\{\varphi_1|\tau(\varphi_2)\} = \psi\varphi_1 \cong \psi\varphi'_1 = \{\varphi'_1|\tau(\varphi_2)\}$. Since by (iv) again, $\tau(\varphi_2) \cong \varphi_2$, and $\varphi_2 \cong \varphi'_2$ by assumption, $\tau(\varphi_2) \cong \varphi'_2$ holds, and applying (ii) in a similar fashion as before, we obtain $\{\varphi'_1|\tau(\varphi_2)\} \cong \{\varphi'_1|\varphi'_2\}$. Putting all these together, $\{\varphi_1|\varphi_2\} \cong \{\varphi'_1|\varphi'_2\}$.

The following useful propositions are proved in the appendix.

Proposition 1. *Static equivalence $\approx_E$ is a formal indistinguishability relation with respect to the equational theory E.*

Proposition 2. *The intersection of an arbitrary number of formal indistinguishability relations (with respect to the same equational theory E) is a formal indistinguishability relation.*

Proposition 3. *Consider static equivalence as a subset $\tilde{E} \subseteq \mathfrak{F}_\mathrm{c} \times \mathfrak{F}_\mathrm{c}$. If $S \subseteq \tilde{E}$, then there is a unique smallest subset $\langle S\rangle \subseteq \tilde{E}$ containing S, such that $\langle S\rangle$ ($\cong_S$ in infix notation) is a formal indistinguishability relation with respect to E. $\langle S\rangle$ can be generated in the following way: Let*

$$S' := \left\{ \begin{array}{c} (\varphi', \varphi'') \\ \in \\ \mathfrak{F}_\mathrm{c} \times \mathfrak{F}_\mathrm{c} \end{array} \middle| \begin{array}{l} \varphi' = \varphi\{\varphi'_1|\ldots|\varphi'_n\} \text{ and } \varphi'' = \varphi\{\varphi''_1|\ldots|\varphi''_n\} \text{ such that} \\ \mathrm{names}(\varphi) = \emptyset \text{ and for all } i = 1, \ldots, n,\ (\varphi'_i, \varphi''_i) \in S, \text{ or} \\ (\varphi''_i, \varphi'_i) \in S, \text{ or } \varphi''_i =_E \tau_i(\varphi'_i) \text{ for some renaming } \tau_i. \end{array} \right\}.$$

Then $\langle S\rangle$ is the transitive closure of S'.

3 Relating Formal and Computational Models

We now present the computational interpretation of the formal model. Our definition is equivalent to the one given by Baudet et al. [10]; the only difference is that we allow probabilistic as opposed to only deterministic interpretations for the symbols in $\mathcal{F}$, and that we have ordered sorts. To each closed term, the interpretation assigns an ensemble of probability distributions on bit strings. This is a generalization of Abadi and Rogaway's definition in [4].

Given a signature $(\mathfrak{S},\mathcal{F})$, where $\mathfrak{S} = (\mathcal{S},\mathcal{S}',\leq_{\mathcal{S}})$, an $(\mathfrak{S},\mathcal{F})$-*computational algebra* $\mathfrak{A}$ is a triple $\mathfrak{A} = (\{\overline{\llbracket s\rrbracket}_{\mathfrak{A}}\}_{s\in\mathcal{S}}, \{\llbracket s\rrbracket_{\mathfrak{A}}\}_{s\in\mathcal{S}'}, \{f_{\mathfrak{A}}\}_{f\in\mathcal{F}})$ as follows: For each sort s in $\mathcal{S}$, $\overline{\llbracket s\rrbracket}_{\mathfrak{A}} = \{\overline{\llbracket s\rrbracket}_{\mathfrak{A}_\eta}\}_\eta$ where $\overline{\llbracket s\rrbracket}_{\mathfrak{A}_\eta} \subseteq \{0,1\}^*$ such that checking whether a bit string is in $\overline{\llbracket s\rrbracket}_{\mathfrak{A}}$ is computable in polynomial-time; for each s in $\mathcal{S}'$, $\llbracket s\rrbracket_{\mathfrak{A}} = \{\llbracket s\rrbracket_{\mathfrak{A}_\eta}\}_\eta$, each $\llbracket s\rrbracket_{\mathfrak{A}_\eta}$ being a probability distribution on $\{0,1\}^*$ with $\mathrm{supp}(\llbracket s\rrbracket_{\mathfrak{A}_\eta}) = \overline{\llbracket s\rrbracket}_{\mathfrak{A}}$ such that there is a polynomial time algorithm to draw random elements from $\llbracket s\rrbracket_{\mathfrak{A}_\eta}$; for each $f \in \mathcal{F}$, with arity $s_1 \times \cdots \times s_k \to s$, $f_{\mathfrak{A}} = \{f_{\mathfrak{A}_\eta}\}_\eta$, where $f_{\mathfrak{A}_\eta} : \overline{\llbracket s_1\rrbracket}_{\mathfrak{A}_\eta} \times \cdots \times \overline{\llbracket s_k\rrbracket}_{\mathfrak{A}_\eta} \to \overline{\llbracket s\rrbracket}_{\mathfrak{A}_\eta}$ is a probabilistic function computable in polynomial time. Here, supp denotes the support of a probability distribution, which is the set where the distribution gives non-zero probability.

Once a computational algebra is fixed, we can associate a probability distribution to each closed term M through the following two algorithms. The interpretation of M is $\mathrm{INTERPRET}_\eta(M)$, which makes use of the algorithm $\mathrm{CONVERT}^\lambda_\eta(M)$ converting M to a tuple of values in $\mathfrak{A}_\eta$ whenever a function $\lambda\colon \mathrm{names}(M) \to \mathrm{supp}(\llbracket \mathfrak{s}(a)\rrbracket_{\mathfrak{A}_\eta})$ is given:

```
algorithm $\mathrm{CONVERT}^\lambda_\eta(M)$
  if $M = a$ with name $a$ then
    return $\lambda(a)$
  if $M = f(M_1,\ldots,M_k)$ then
    for $i = 1,\ldots,k$ do
    $e_i \xleftarrow{R} \mathrm{CONVERT}^\lambda_\eta(M_i)$
    $v \xleftarrow{R} f_{\mathfrak{A}_\eta}(e_1,\ldots,e_k)$
    return $v$
```

```
algorithm $\mathrm{INTERPRET}_\eta(M)$
  for $a \in \mathrm{names}(M)$ do
    $\lambda(a) \xleftarrow{R} \llbracket \mathfrak{s}(a)\rrbracket_{\mathfrak{A}_\eta}$
    $v \xleftarrow{R} \mathrm{CONVERT}^\lambda_\eta(M)$
  return $v$

algorithm $\mathrm{INTERPRET}'_\eta(\varphi)$
  for $a \in \mathrm{names}(\varphi)$ do $\lambda(a) \xleftarrow{R} \llbracket \mathfrak{s}(a)\rrbracket_{\mathfrak{A}_\eta}$
  if $\varphi = \{x_1 = T_1,\ldots,x_n = T_n\}$ then
    $e_i \xleftarrow{R} \mathrm{CONVERT}^\lambda_\eta(T_i)$  $i = 1,\ldots,n$
    $v \longleftarrow \{x_1 = e_1,\ldots,x_n = e_n\}$
    return $v$
```

For each η, the probability distribution of $v \xleftarrow{R} \mathrm{CONVERT}_\eta(M)$ is denoted by $\llbracket M\rrbracket_{\mathfrak{A}_\eta}$. The ensemble $\{\llbracket M\rrbracket_{\mathfrak{A}_\eta}\}_\eta$ is denoted by $\llbracket M\rrbracket_{\mathfrak{A}}$. We call $\llbracket M\rrbracket_{\mathfrak{A}}$ the *computational interpretation* of the term M. For any name $a\colon s$ with $s \in \mathcal{S}'$, $\llbracket s\rrbracket_{\mathfrak{A}} = \llbracket \mathfrak{s}(a)\rrbracket_{\mathfrak{A}} = \llbracket a\rrbracket_{\mathfrak{A}}$. We define the interpretation of a closed frame $\varphi = \{x_1 = T_1,\ldots,x_n = T_n\}$ via the algorithm $\mathrm{INTERPRET}'_\eta(\varphi)$. We use $\llbracket \varphi\rrbracket_{\mathfrak{A}_\eta}$ for the probability distribution given by $\mathrm{INTERPRET}'_\eta(\varphi)$ and $\llbracket \varphi\rrbracket_{\mathfrak{A}}$ for the ensemble of these distributions, which we call the *computational interpretation* of the frame φ in the model $\mathfrak{A}$.

Two ensembles of probability distributions are said to be *computationally indistinguishable*, if no probabilistic polynomial time algorithm can distinguish them. Once the formal expressions are interpreted, then we can consider the computational indistinguishability of interpretations of two closed terms or two closed frames. We will use the notation $[\![M_1]\!]_{\mathfrak{A}} \approx [\![M_2]\!]_{\mathfrak{A}}$ and $[\![\varphi_1]\!]_{\mathfrak{A}} \approx [\![\varphi_2]\!]_{\mathfrak{A}}$, respectively. Explicitly, this latter means that for any PPT algorithm $\mathcal{A}$,

$$\left|\Pr[\widehat{\varphi_1} \xleftarrow{R} [\![\varphi_1]\!]_{\mathfrak{A}_\eta} : \mathcal{A}(\eta, \widehat{\varphi_1}) = 1] - \Pr[\widehat{\varphi_2} \xleftarrow{R} [\![\varphi_2]\!]_{\mathfrak{A}_\eta} : \mathcal{A}(\eta, \widehat{\varphi_2}) = 1]\right|,$$

denoted by $\mathrm{Adv}^{\mathcal{A}}_{\eta}([\![\varphi_1]\!]_{\mathfrak{A}}, [\![\varphi_2]\!]_{\mathfrak{A}})$, is a negligible function; that is, for each $n \in \mathbb{N}$ and all sufficiently large η, $\mathrm{Adv}^{\mathcal{A}}_{\eta}([\![\varphi_1]\!]_{\mathfrak{A}}, [\![\varphi_2]\!]_{\mathfrak{A}}) < \eta^{-n}$.

4 Soundness, Completeness and Faithfulness

The computational model of a cryptographic scheme is in a sense closer to reality than its formal representation by being a more detailed description. Therefore, the accuracy of a formal model can be characterized based on how close it is to the computational model; more specifically, how formal and computational indistinguishability relate to each other via the interpretation. The most important concepts to describe this are given in the following definition.

Definition 4. *Let $\mathfrak{A}$ be an $(\mathfrak{S}, \mathcal{F})$-computational algebra, and let $\cong$ be a formal indistinguishability relation on the set of frames, and let $F \subseteq \mathfrak{F}_{\mathrm{c}}$. We say that the computational algebra $\mathfrak{A}$ is* $\cong$-sound *on F if for every closed pair of frames $\varphi_1, \varphi_2 \in F$, $\varphi_1 \cong \varphi_2$ implies that $[\![\varphi_1]\!]_{\mathfrak{A}} \approx [\![\varphi_2]\!]_{\mathfrak{A}}$. $\mathfrak{A}$ is* $\cong$-complete *on F if for every closed pair of frames $\varphi_1, \varphi_2 \in F$, $\varphi_1 \ncong \varphi_2$ implies that $[\![\varphi_1]\!]_{\mathfrak{A}} \not\approx [\![\varphi_2]\!]_{\mathfrak{A}}$.* $\mathfrak{A}$ is $\cong$-faithful *on F if for every closed pair of frames $\varphi_1, \varphi_2 \in F$, $\varphi_1 \ncong \varphi_2$ implies that the statistical distance $\Delta([\![\varphi_1]\!]_{\mathfrak{A}_\eta}, [\![\varphi_2]\!]_{\mathfrak{A}_\eta})$ is not negligible and there is a PPT algorithm $\mathcal{A}$ such that $|\,\mathrm{Adv}^{\mathcal{A}}_{\eta}([\![\varphi_1]\!]_{\mathfrak{A}_\eta}, [\![\varphi_2]\!]_{\mathfrak{A}_\eta}) - \Delta([\![\varphi_1]\!]_{\mathfrak{A}_\eta}, [\![\varphi_2]\!]_{\mathfrak{A}_\eta})|$ is negligible. For all three notions, we adopt the convention that if no such set F is mentioned, it is assumed that $F = \mathfrak{F}_{\mathrm{c}}$.*

It is well known that the advantage of an adversary trying to distinguish two distributions is less than or equal to the statistical distance between the two distributions. Faithfulness therefore means that if two frames are formally distinguishable, then there is an algorithm that distinguishes their interpretations almost optimally.

Remark 2. If a model is sound, then formal proofs of indistinguishability are valid proofs of computational indistinguishability.

Our faithfulness definition is different from the one by Baudet et al. given in [10]. They require the existence of an adversary whose advantage is negligibly close to 1. However, there are interesting cases where their requirement is too strong, as the following example shows. Nevertheless, we will not discuss faithfulness in this paper beyond this example.

Example 1. Suppose that we add a function symbol LSB: $Data \to Data$ to our theory, where *Data* is a sort. We think of this as the least significant bit, and accordingly, we define the interpretation for the LSB function such that $\mathrm{LSB}_{\mathfrak{A}_\eta}(x)$ is the least significant bit of x. Suppose that names of sort *Data* get interpreted as bit strings with a certain maximum length with uniform distribution. Now, consider the two frames $\nu ab.\{x_1 = \mathrm{LSB}(a), x_2 = \mathrm{LSB}(b)\}$ and $\nu a.\{x_1 = \mathrm{LSB}(a), x_2 = \mathrm{LSB}(a)\}$. After interpretation, for each security parameter, the first frame will result in two independent bits of uniform distribution, whereas the interpretation of the second frame will contain two completely correlated bits of uniform distribution. No adversary can distinguish these two distributions with advantage greater than 1/2, which is the statistical distance. However, the adversary that outputs 1 if the two bits are identical and 0 if they are different is clearly the best possible.

Remark 3. Completeness can be rewritten in the form that for every closed pair of frames φ_1, φ_2, $[\![\varphi_1]\!]_{\mathfrak{A}} \approx [\![\varphi_2]\!]_{\mathfrak{A}}$ implies $\varphi_1 \cong \varphi_2$. This notion is weaker then faithfulness, i.e., a faithful interpretation is also complete.

Let us introduce some notation. Let $\mathfrak{A}$ be an $(\mathfrak{S}, \mathcal{F})$-computational algebra, and let $\{x_1 = T_1, x_2 = T_2\}$ be a closed frame in this setting. Then by $e_1, e_2 \xleftarrow{R} [\![T_1, T_2]\!]_{\mathfrak{A}_\eta}$, we will denote the random sampling $\{x_1 = e_1, x_2 = e_1\} \xleftarrow{R} \mathrm{INTERPRET}'_\eta(\{x_1 = T_1, x_2 = T_2\})$.

Definition 5. *Let $\mathfrak{A}$ be a $(\mathfrak{S}, \mathcal{F})$-computational algebra, and let E be an equational theory. We say that $\mathfrak{A}$ is* $=_E$-sound *if for each pair of closed terms T_1 and T_2, $T_1 =_E T_2$ implies that* $\Pr[e_1, e_2 \xleftarrow{R} [\![T_1, T_2]\!]_{\mathfrak{A}_\eta} \colon e_1 \neq e_2]$ *is negligible. It is* $=_E$-complete *if for each pair of closed terms T_1 and T_2, $T_1 \neq_E T_2$ implies that* $\Pr[e_1, e_2 \xleftarrow{R} [\![T_1, T_2]\!]_{\mathfrak{A}_\eta} \colon e_1 \neq e_2]$ *is not negligible.*

Remark 4. The reader may ask why no adversaries are used in this definition. For example, would it not make more sense to define $=_E$-soundness so that for each pair of closed terms T_1 and T_2 if $T_1 =_E T_2$ holds, then $[\![T_1, T_2]\!]_{\mathfrak{A}} \approx [\![T_1, T_1]\!]_{\mathfrak{A}}$? However, using the fact that the advantage of an adversary trying to distinguish the two distributions cannot exceed the statistical distance, it is easy to show that this definition would be equivalent to what is given above.

The following proposition shows that if a FIR $\cong$ is generated by a set $S \subseteq \mathfrak{F}_c \times \mathfrak{F}_c$, then it suffices to check soundness for pairs of frames in S to see that $\langle S \rangle$ is sound. The proof is included in the appendix.

Proposition 4. *Let $\mathfrak{A}$ be an $(\mathfrak{S}, \mathcal{F})$-computational algebra that is $=_E$-sound. Suppose $S \subseteq \tilde{E}$ is a binary relation on closed frames such that $(\varphi, \psi) \in S$ implies $[\![\varphi]\!]_{\mathfrak{A}_\eta} \approx [\![\psi]\!]_{\mathfrak{A}_\eta}$. Then $[\![\varphi]\!]_{\mathfrak{A}_\eta} \approx [\![\psi]\!]_{\mathfrak{A}_\eta}$ whenever $\varphi \cong_S \psi$. That is, $\mathfrak{A}$ is $\cong_S$-sound.*

Corollary 1. *Let $\mathfrak{A}$ be a $(\mathfrak{S}, \mathcal{F})$-computational algebra with $S \subseteq \mathfrak{F}_c \times \mathfrak{F}_c$ such that $\mathfrak{A}$ is $\cong_S$-sound on F. Let $S' := S \cap (F \times F)$. Then $\mathfrak{A}$ is $\cong_{S'}$-sound.*

5 Applications

In this section, we exemplify the utility of formal indistinguishability relations that refine static equivalence. In the theory of groups with exponentiation, we obtain an FIR such that soundness is equivalent to the Decisional Diffie-Hellman Assumption. Next, we show how to handle key-cycles by encoding Laud's approach in a formal indistinguishability relation. Finally, we give an example from propositional Boolean logic whose natural model would not be sound with respect to static equivalence, but is sound with respect to a particular FIR.

5.1 Decisional Diffie-Hellman Assumption

Consider the following equational theory to model a commutative group with exponentiation (as in [10]). Let A and G be sorts, $\mathcal{S} = \mathcal{S}'$ with the trivial ordering, and let $\mathcal{F}$ contain the following function symbols: $*\colon G \times G \to G$; $1_G\colon G$; $\cdot\colon A \times A \to A$; $+\colon A \times A \to A$; $-\colon A \to A$; $1_A\colon A$; $0\colon A$; $\exp\colon G \times A \to G$. To simplify our notation, we write U^V for $\exp(U, V)$.

Let the equational theory E be generated by the following equations:

$$\begin{array}{ccc}
x * 1_G = x & a + (-a) = 0 & (a+b) \cdot c = a \cdot c + b \cdot c \\
x * y = y * x & a + (b + c) = (a + b) + c & (x^a)^b = x^{(a \cdot b)} \\
x * (y * z) = (x * y) * z & a \cdot 1_A = a & x^a * x^b = x^{a+b} \\
a + 0 = a & a \cdot b = b \cdot a & x^{1_A} = x \\
a + b = b + a & a \cdot (b \cdot c) = (a \cdot b) \cdot c & x^0 = 1_G
\end{array}$$

Observe that we did not include a symbol for the discrete logarithm in the language. The reason is that we want to assume that computing a from g^a is not feasible for an adversary.

Once a computational group scheme is set (for computational group schemes see for example the full version of [13]), the computational interpretation of this signature is straightforward. Names of sort G will be mapped to the ensemble of distributions corresponding to the generation of random group elements whereas names of sort A will correspond to the generation of ring elements. Addition, multiplication etc. will be translated to addition, multiplication etc. of ring or group elements, respectively. As Baudet et al. point out in their paper, in this theory, the frames $\nu g, a, b.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = g^{ab}\}$ and $\nu g, a, b, c.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = g^c\}$ are statically equivalent. Distinguishing the interpretations of these two frames is the Decisional Diffie-Hellman problem. So, a computational implementation that is sound with respect to static equivalence will imply that the DDH assumption holds for the given group scheme. Unfortunately, soundness would imply much more than the DDH assumption. For example, $\nu gab.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = g^{a^n b^m}\} \approx_E \nu gabc.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = g^c\}$ for some naturals $n, m \geq 1$, and therefore $\approx_E$-soundness would imply that the computational interpretations of these are indistinguishable as well. Moreover, even $\nu gab.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = ab\} \approx_E \nu gabc.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = c\}$. It is unreasonable to require that all these hold for a computational implementation.

We therefore suggest to use a formal indistinguishability relation instead. Since we only want to assume that the DDH assumption holds and nothing more, simply let S be the set consisting of the pair

$$\big(\nu gab.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = g^{ab}\}, \nu gabc.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = g^c\}\big)\ .$$

Then, by Proposition 4, a computational interpretation is $\cong_S$-sound if and only if the DDH assumption holds. In this model, $\cong_S$ will make exactly those frames equivalent for which equivalence necessarily follows from the DDH assumption and the algebraic identities that we included in the model. Hence, for example, $\nu gab.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = ab\} \not\approx_E \nu gabc.\{x_1 = g, x_2 = g^a, x_3 = g^b, x_4 = c\}$, but $\nu gg'ab.\{x_1 = gg', x_2 = (gg')^a, x_3 = (gg')^b, x_4 = (gg')^{ab}\} \cong_S \nu gg'c.\{x_1 = gg', x_2 = (gg')^a, x_3 = (gg')^b, x_4 = (gg')^c\}$. This follows from the commutativity of the group operation, from property (ii) and (iv) of the formal indistinguishability relation, and the definition of S.

Often (for example in the case of uniform distributions), $[\![\{x_1 = g, x_2 = g'\}]\!]_{\mathfrak{A}}$ and $[\![\{x_1 = g, x_2 = gg'\}]\!]_{\mathfrak{A}}$ are computationally indistinguishable. In this case, we can include the pair of frames $(\nu gg'.\{x_1 = g, x_2 = g'\}, \nu gg'.\{x_1 = g, x_2 = gg'\})$ in S, and then the above equivalence will follow without the use of commutativity. Alternatively, if we include $(\nu ab.\{x_1 = a, x_2 = b\}, \nu ab.\{x_1 = a, x_2 = ab\})$ in S, then it follows that $\nu gg'adf.\{x_1 = g, x_2 = g^{ad}, x_3 = g^f, x_4 = g'g^{adf}\} \cong_S \nu gg'df.\{x_1 = g, x_2 = g^{ad}, x_3 = g^f, x_4 = g'g^c\}$.

5.2 Key-Cycles

In their paper, Baudet et al. [10] also consider an equational theory for encryption schemes, and prove the soundness of static equivalence when key-cycles are excluded. Another such example in the framework of static equivalence can be found in the paper of Abadi et al. [1], where key-cycles are excluded as well. The problem of key-cycles is not specific to static equivalence. It necessarily comes up in the investigation of the relationship of formal and computational models; already Abadi and Rogaway in [4] had to exclude key-cycles. There are two ways to include them: Either the encryption scheme used for the interpretation has to be secure even in the presence of key-cycles, or the formal indistinguishability notion has to be relaxed. The problem with the former is that no realistic encryption scheme is known to be secure for key-cycles. Laud proposed a simple solution pursuing the second approach in [18]: Simply assume that the formal adversary can decrypt all the ciphertexts that were encrypted by keys that are part of a key-cycle. In the present formalism this means switching from static equivalence to another indistinguishability relation. We illustrate this by first recasting the original Abadi-Rogaway treatment into the present formalism and then showing how Laud's solution provides a special FIR.

The Abadi-Rogaway formal language of [4] gives a signature $(\mathfrak{S}_{\mathrm{senc}}, \mathcal{F}_{\mathrm{senc}})$, where $\mathfrak{S}_{\mathrm{senc}} = (\mathcal{S}_{\mathrm{senc}}, \mathcal{S}'_{\mathrm{senc}}, \leq_{\mathcal{S}})$, with $\mathcal{S}_{\mathrm{senc}} = \{\mathit{Key}, \mathit{Data}, \mathit{Cipher}, \mathit{Pair}\}$, $\mathcal{S}'_{\mathrm{senc}} = \{\mathit{Key}\}$, $\mathit{Key} \leq_{\mathcal{S}} \mathit{Data}$, $\mathit{Cipher} \leq_{\mathcal{S}} \mathit{Data}$, $\mathit{Pair} \leq_{\mathcal{S}} \mathit{Data}$. The following function symbols are in $\mathcal{F}_{\mathrm{senc}}$:

$\mathbf{enc}$	:	$\mathit{Data} \times \mathit{Key} \rightarrow \mathit{Cipher}$	symmetric encryption
$\mathbf{dec}$	:	$\mathit{Data} \times \mathit{Data} \rightarrow \mathit{Data}$	symmetric decryption
$\mathbf{pair}$	:	$\mathit{Data} \times \mathit{Data} \rightarrow \mathit{Pair}$	pairing
$\mathbf{fst}$	:	$\mathit{Data} \rightarrow \mathit{Data}$	first projection
$\mathbf{snd}$	:	$\mathit{Data} \rightarrow \mathit{Data}$	second projection
$0, 1, \mathrm{error}$	:	Data	constants

Let the equational theory E_{senc} be generated by the following equations: $\mathbf{dec}(\mathbf{enc}(x, y), y) = x$, $\mathbf{fst}(\mathbf{pair}(x, y)) = x$, $\mathbf{snd}(\mathbf{pair}(x, y)) = y$, $\mathbf{pair}(\mathbf{fst}(x), \mathbf{snd}(x)) = x$, and furthermore, $\mathbf{dec}(x, y) = \mathrm{error}$ whenever the sort $\mathfrak{s}(x)$ is $\leq_{\mathcal{S}}$-incomparable with *Cipher* or $\mathfrak{s}(y)$ is incomparable with *Key*, and $\mathbf{fst}(x) = \mathbf{snd}(x) = \mathrm{error}$ whenever $\mathfrak{s}(x)$ is incomparable with *Pair*.

Given a computational encryption scheme $(\mathcal{E}, \mathcal{D}, \mathcal{K})$, along with a computational way of pairing, it is straightforward how to assign a computational algebra $\mathfrak{A}_{\mathrm{senc}}$ to this signature: Simply interpret the formal function symbols as their computational counterpart, and let $[\![\mathit{Key}]\!]_{\mathfrak{A}_{\mathrm{senc}}}$ be the distribution of key generation.

Definition 6. *A frame φ is* well-formed *if φ does not contain the symbols* $\mathbf{dec}$*,* $\mathbf{fst}$*,* $\mathbf{snd}$*,* error*. For a well-formed frame φ, the set of* recoverable keys *of φ are those keys that are deducible from φ, i. e., $\mathit{R\text{-}Keys}(\varphi) = \{k \mid k \in \mathrm{names}(\varphi),\ k\colon \mathit{Key},\ \varphi \vdash_{E_{\mathrm{senc}}} k\}$. The set $\mathit{B\text{-}Keys}(\varphi)$ consists of those keys that encrypt the outermost undecryptable terms in φ, namely, those undecryptable terms that are deducible from φ:*

$$\mathit{B\text{-}Keys}(\varphi) = \{k \in \mathrm{names}(\varphi) \mid \varphi \vdash_{E_{\mathrm{senc}}} \mathbf{enc}(T, k) \text{ for some } T, \text{ and } k \notin \mathit{R\text{-}Keys}(\varphi)\}$$

We say that $\mathit{B\text{-}Keys}(\varphi)$ is cyclic *in φ, if for some keys $k_1, k_2, \ldots, k_m \in \mathit{B\text{-}Keys}(\varphi)$ with $k_1 = k_m$, there are terms $M_1, \ldots, M_m$ with $M_1 = M_m$ such that k_i occurs in M_i in positions other than in the second argument of* $\mathbf{enc}$ *and $\varphi \vdash_{E_{\mathrm{senc}}} \mathbf{enc}(M_i, k_{i+1})$.*

The reason for excluding some symbols from well-formed frames is that Abadi and Rogaway only considered expressions built via encryption and pairing. But these symbols of course can be used in the distinguishers M and N in the definition of static equivalence! The result of Abadi and Rogaway then says that if the encryption scheme is type-0 secure (as defined in [4]), then for two *well-formed* frames, $\varphi \approx_{E_{\mathrm{senc}}} \psi$ implies $[\![\varphi]\!]_{\mathfrak{A}_{\mathrm{senc}}} \approx [\![\psi]\!]_{\mathfrak{A}_{\mathrm{senc}}}$ whenever neither $\mathit{B\text{-}Keys}(\varphi)$ nor $\mathit{B\text{-}Keys}(\psi)$ are cyclic in the corresponding frames..

The exclusion of key-cycles is necessary if the encryption scheme is just type-0 secure. In fact, all standard computational notions of security make it necessary to exclude key-cycles. If the encryption scheme satisfies stronger security definitions, for instance if it is KDM-secure (see [5]), then key-cycles do not cause problems, but no realistic KDM-secure encryptions are known at this time.

As we mentioned, following Laud's method, we can keep the computational algebra $\mathfrak{A}_{\mathrm{senc}}$ but switch from static equivalence to another formal indistinguishability relation on the formal side which is sound even in the presence of key-cycles. Define S as static equivalence $\tilde{E}_{\mathrm{senc}}$ minus pairs that contain key-cycles on at least one side. Then $\cong_S$-soundness including key-cycles will hold. More precisely, the following proposition is true:

Proposition 5. *Let $\mathfrak{A}_{\text{senc}}$ be the above $(\mathfrak{S}_{\text{senc}}, \mathcal{F}_{\text{senc}})$-computational algebra. Let $S \subseteq \mathfrak{F}_{\text{c}} \times \mathfrak{F}_{\text{c}}$ be the following set:*

$$S := \Big\{ (\varphi_1, \varphi_2) \,\Big|\, (\varphi_1, \varphi_2) \in \tilde{E}_{\text{senc}} \text{ and, if } \varphi_i \text{ is well-formed, } B\text{-}Keys(\varphi_i) \text{ is not cyclic} \Big\}.$$

Let $\cong_S$ be the formal indistinguishability relation generated by S. Then, for all well-formed frames φ_1 and φ_2, $\varphi_1 \cong_S \varphi_2$ implies $[\![\varphi_1]\!]_{\mathfrak{A}_{\text{senc}}} \approx [\![\varphi_2]\!]_{\mathfrak{A}_{\text{senc}}}$.

The proposition clearly holds on S, because from there we removed the the key-cycles. Then the proof is similar to that of Proposition 4.

5.3 Boolean Algebra

We give an example where static equivalence identifies frames that are computationally clearly distinguishable, whereas a more fine-grained formal indistinguishability relation can do better.

Consider a signature $(\mathfrak{S}, \mathcal{F})$, where $\mathfrak{S} = (\{B, S\}, \{B, S\}, =)$, and $\mathcal{F}$ contains the symbols $\wedge, \vee \colon B \to B$, constants $0, 1 \colon B$, as well as $\text{LSB} \colon S \to B$. Let E be the equational theory generated by the set $\{(M, N) \mid M, N \colon B,\ M \leftrightarrow N \text{ is a tautology of propositional Boolean algebra}\}$.

Let $\mathfrak{A}$ denote the following $(\mathfrak{S}, \mathcal{F})$-computational algebra: $\text{supp}([\![S]\!]_{\mathfrak{A}_\eta}) = \{0,1\}^\eta \subset \{0,1\}^*$, $\text{supp}([\![B]\!]_{\mathfrak{A}_\eta}) = \{0,1\} \subset \{0,1\}^*$, where both spaces are equipped with the uniform distribution over their support. The operations 0, 1, $\wedge$, $\vee$ are interpreted as the obvious operations on the Boolean algebra $\{0,1\}$, and $\text{LSB}_{\mathfrak{A}_\eta}$ is defined by $\text{LSB}_{\mathfrak{A}_\eta}(b_1 \ldots b_\eta) = b_\eta$. It is clear that $\mathfrak{A}$ is $=_E$-sound. However, it is not $\approx_E$-sound because

$$\nu ab.\{x = \text{LSB}(a) \wedge \text{LSB}(b)\} \approx_E \nu cd.\{x = \text{LSB}(c) \vee \text{LSB}(d)\}$$

if $a, b : S$, or, even more simply, $\nu ab.\{x = a \wedge b\} \approx_E \nu cd.\{x = c \vee d\}$ for $a, b : B$, whereas the interpretation of the left-hand side is distributed so that $\Pr[\{x = 1\}] = 1/4$, and for the right-hand side $\Pr[\{x = 1\}] = 3/4$, which are clearly distinguishable. (We remark that while $\mathfrak{A}$ does not satisfy a requirement of Baudet et al. that, for two names $a, b \colon B$, $\Pr[e_1, e_2 \leftarrow [\![a, b]\!]_{\mathfrak{A}_\eta}; e_1 = e_2]$ be negligible, this can also be satisfied by making minor changes to the model.)

To remedy the problem, we can use instead of static equivalence a custom formal indistinguishability relation. For a frame which has only sort B in its domain, it is easy to compute explicitly the probability distribution of its interpretation using only the formal expressions. Without writing down the explicit recursive formula, just consider for example that for $\nu ab.\{x_1 = a \wedge b, x_2 = a\}$, $\Pr[\{x_1 = 1, x_2 = 1\}] = 1/4$, $\Pr[\{x_1 = 1, x_2 = 0\}] = 0$, $\Pr[\{x_1 = 0, x_2 = 1\}] = 1/4$, $\Pr[\{x_1 = 0, x_2 = 0\}] = 1/2$. We can therefore define the binary relation S generating the FIR so that S contains those pairs for which the domains only have variables of sort B, and have identical probability distributions. This definition gives a formal indistinguishability relation that is both sound and faithful.

6 Conclusion

We suggested a generalized notion of formal indistinguishability which provides greater flexibility than static equivalence. This is needed because computational distinguishability is much more than just trying to distinguish with the algebraic manipulations allowed by the formal model. It is unrealistic to expect that an indistinguishability relation defined in a purely algebraic manner in a relatively simple formal model will cover all subtleties of computational indistinguishability. However, even though computational indistinguishability is a complex notion, in many cases it is possible to distill a simple formal indistinguishability relation, impose it on the formal model, and get a sound, meaningful theory. The utility of this new definition was demonstrated in Section 5: We pointed out natural models of certain equational theories in which static equivalence seems to be an insufficiently coarse notion of formal indistinguishability, and showed how to come up with different indistinguishability relations that do not identify more expressions than needed.

References

1. M. Abadi, M. Baudet, and B. Warinschi. Guessing attacks and the computational soundness of static equivalence. In L. Aceto and A. Ingólfsdóttir, editors, *Proceedings of the 9th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS '06)*, volume 3921 of *Lecture Notes in Computer Science*, pages 398–412. Springer-Verlag, March-April 2006.
2. M. Abadi and C. Fournet. Mobile values, new names, and secure communication. In *POPL '01: Proceedings of the 28th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 104–115, New York, NY, USA, 2001. ACM Press.
3. M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The Spi Calculus. *Information and Computation*, 148(1):1–70, January 1999. Full version available as SRC Research Report 149, January 1998.
4. M. Abadi and P. Rogaway. Reconciling two views of cryptography (the computational soundness of formal encryption). *Journal of Cryptology*, 15(2):103–127, January 2002.
5. P. Adão, G. Bana, J. Herzog, and A. Scedrov. Soundness of formal encryption in the presence of key-cycles. In S. De Capitani di Vimercati, P. Syverson, and D. Gollmann, editors, *Proceedings of the 10th European Symposium on Research in Computer Security (ESORICS)*, volume 3679 of *Lecture Notes in Computer Science*, pages 374–396, Milan, Italy, September 12–14 2005. Springer.
6. P. Adão, G. Bana, and A. Scedrov. Computational and information-theoretic soundness and completeness of formal encryption. In *Proceedings of the 18th IEEE Computer Security Foundations Workshop (CSFW)*, pages 170–184, Aix-en-Provence, France, June 20–22 2005. IEEE Computer Society Press.
7. M. Backes and B. Pfitzmann. Symmetric encryption in a simulatable Dolev-Yao style cryptographic library. In *Proceedings of the 17th IEEE Computer Security Foundations Workshop (CSFW)*, pages 204–218, Pacific Grove, CA, USA, June 28–30 2004. IEEE Computer Society Press. Full version available at IACR ePrint Archive, Report 2004/059.

8. M. Backes, B. Pfitzmann, and M. Waidner. A composable cryptographic library with nested operations. In S. Jajodia, V. Atluri, and T. Jaeger, editors, *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)*, pages 220–230, Washington D.C., USA, October 27–30 2003. ACM Press. Full version available at IACR ePrint Archive, Report 2003/015, January 2003.
9. Gergei Bana, Payman Mohassel, and Till Stegers. Computational soundness of formal indistinguishability and static equivalence. In *Proceedings of ASIAN 2006*, Lecture Notes in Computer Science. Springer-Verlag, 2007. To appear.
10. M. Baudet, V. Cortier, and S. Kremer. Computationally sound implementations of equational theories against passive adversaries. In *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP'05)*, volume 3580, pages 652–663. Springer-Verlag, July 2005.
11. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 136–145, Las Vegas, NV, USA, October 14–17 2001. IEEE Computer Society Press. Full version available at IACR ePrint Archive, Report 2000/067.
12. Ran Canetti and Jonathan Herzog. Universally composable symbolic analysis of mutual authentication and key exchange protocols. In *Proceedings, Theory of Cryptography Conference (TCC)*, March 2006.
13. R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In H. Krawczyk, editor, *Advances in Cryptology CRYPTO '98*, volume 1462 of *Lecture Notes in Computer Science*, pages 13–25, Santa Barbara, CA, USA, August 23–27 1998. Springer. Full version available at IACR ePrint Archive, Report 2001/108, Dec 2001.
14. A. Datta, A. Derek, J. C. Mitchell, V. Shmatikov, and M. Turuani. Probabilistic polynomial-time semantics for a protocol security logic. In L. Caires, G. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, editors, *Proceedings of the The 32nd International Colloquium on Automata, Languages and Programming (ICALP)*, volume 3580 of *Lecture Notes in Computer Science*, pages 16–29, Lisbon, Portugal, July 11–15 2005. Springer.
15. D. Dolev and A. C. Yao. On the security of public-key protocols. *IEEE Transactions on Information Theory*, 29(2):198–208, March 1983.
16. S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and Systems Sciences*, 28(2):270–299, April 1984.
17. J. D. Guttman, F. J. Thayer, and L. D. Zuck. The faithfulness of abstract protocol analysis: Message authentication. In P. Samarati, editor, *Proceedings of the 8th ACM Conference on Computer and Communications Security (CCS)*, pages 186–195, Philadelphia, PA, USA, November 05–08 2001. ACM Press.
18. P. Laud. Encryption cycles and two views of cryptography. In *Proceedings of the 7th Nordic Workshop on Secure IT Systems (NORDSEC)*, number 31 in Karlstad University Studies, pages 85–100, Karlstad, Sweden, November 7–8 2002.
19. P. Laud. Symmetric encryption in automatic analyses for confidentiality against active adversaries. In *Proceedings of the 2004 IEEE Symposium on Security and Privacy (S&P)*, pages 71–85, Oakland, CA, USA, May 9–12 2004. IEEE Computer Society Press.
20. D. Micciancio and B. Warinschi. Completeness theorems for the Abadi-Rogaway logic of encrypted expressions. *Journal of Computer Security*, 12(1):99–130, 2004.

A Proofs

A.1 Proof of Proposition 1

Proof. Items (i) and (iii) are trivially satisfied by $\approx_E$. Consider frames $\varphi, \varphi_1, \varphi_2$ as in (ii). Let M, N be terms whose variables are included in $\mathrm{dom}(\varphi\varphi_1) = \mathrm{dom}(\varphi\varphi_2) = \mathrm{dom}(\varphi)$ and that have no names in common with $\varphi\varphi_i$, $i = 1, 2$. Then $\mathrm{names}(M\varphi) = \mathrm{names}(M) \cup \mathrm{names}(\varphi)$, and $\mathrm{names}(M) \cup \mathrm{names}(\varphi)$ is disjoint from $\mathrm{names}(\varphi_i)$ by the assumption on M and condition (ii). Therefore $\mathrm{names}(M\varphi)$ and $\mathrm{names}(\varphi_i)$ are disjoint (and likewise for N). If $\varphi_1 \approx_E \varphi_2$ holds, then by the definition of static equivalence, $(M\varphi)\varphi_1 =_E (N\varphi)\varphi_1$ if and only if $(M\varphi)\varphi_2 =_E (N\varphi)\varphi_2$. Therefore, $M(\varphi\varphi_1) =_E N(\varphi\varphi_1)$ if and only if $M(\varphi\varphi_2) =_E N(\varphi\varphi_2)$, and that is exactly what we had to prove to see that $\varphi\varphi_1 \cong \varphi\varphi_2$.

To see (iv), we first construct another renaming τ' the following way: On $\mathrm{names}(\varphi)$, let τ' be equal τ, and on $\mathcal{N} \setminus (\mathrm{names}(\varphi) \cup \tau(\mathrm{names}(\varphi))$, let τ' be the identity map. We still have to define τ' on $\tau(\mathrm{names}(\varphi)) \setminus \mathrm{names}(\varphi)$. Since τ is a sort-preserving bijection, the number of elements in $[\tau(\mathrm{names}(\varphi)) \setminus \mathrm{names}(\varphi)]_s$ is the same as the number of elements in $[\mathrm{names}(\varphi) \setminus \tau(\mathrm{names}(\varphi))]_s$ for each sort s; both are $|[\mathrm{names}(\varphi)]_s| - |[\mathrm{names}(\varphi) \cap \tau(\mathrm{names}(\varphi))]_s|$, which equals $|[\tau(\mathrm{names}(\varphi))]_s| - |[\mathrm{names}(\varphi) \cap \tau(\mathrm{names}(\varphi))]_s|$. So on $[\tau(\mathrm{names}(\varphi)) \setminus \mathrm{names}(\varphi)]_s$ choose τ' to be any bijection to $[\mathrm{names}(\varphi) \setminus \tau(\mathrm{names}(\varphi))]_s$. It is then easy to see that τ' is a sort-preserving bijection on $\mathcal{N}$, and that $\tau'(\varphi) = \tau(\varphi)$ for the φ in question. Moreover, for any M expression that shares no names with φ and $\tau(\varphi)$, $\tau'(M) = M$, and therefore $M\tau(\varphi) = M\tau'(\varphi) = \tau'(M\varphi)$ holds. Hence, for any two such expressions M and N, $M\tau(\varphi) =_E N\tau(\varphi)$ if and only if $\tau'(M\varphi) =_E \tau'(N\varphi)$ which happens – since τ' is a bijection – if and only if $M\varphi =_E N\varphi$, and $\tau(\varphi) \cong \varphi$ follows.

A.2 Proof of Proposition 2

Proof. Let $(\cong_i)_{i \in I}$, where I is some indexing set, be a sequence of formal indistinguishability relations with respect to the same equational theory E, and let $\cong$ be their intersection. Clearly, $\cong$ is an equivalence relation. Items (i) and (iii) are trivially satisfied by $\cong$. Let $\varphi, \varphi_1, \varphi_2$ be as in (ii). Then $\varphi_1\varphi \cong_i \varphi_2\varphi$ for all $i \in I$, hence $\varphi_1\varphi \cong \varphi_2\varphi$. Likewise, since every $\cong_i$ is preserved by the renaming of variables, $\cong$ is preserved as well. Therefore (iii), (iv) are also satisfied by $\cong$.

A.3 Proof of Proposition 3

Proof. The existence of such a smallest set is clear. In order to prove the statement about how to generate $\langle S \rangle$, consider the transitive closure $\hat{S}$ of S'. It is clear from the definition of S' that $\hat{S}$ is symmetric, reflexive and transitive, hence an equivalence relation. It is also clear from the definition of a formal indistinguishability relation and from Remark 1 that in the construction of S'

and $\hat{S}$ we stay within $\langle S \rangle$. Therefore, we only have to show that $\hat{S}$ is a formal indistinguishability relation.

By the construction of $\hat{S}$, it is clear that $\hat{S}$ satisfies properties (i), (iii) and (iv) of a formal indistinguishability relation, so that only (ii) remains. Suppose $(\varphi_1, \varphi_2) \in \hat{S}$, and let φ be as in (ii); we have to show that $(\varphi\varphi_1, \varphi\varphi_2) \in \hat{S}$. Since $(\varphi_1, \varphi_2) \in \hat{S}$, there are frames $\psi_1, \ldots, \psi_n$ such that $\varphi_1 = \psi_1$, $\varphi_2 = \psi_n$, and the pairs (ψ_i, ψ_{i+1}), where $i = 1, \ldots, n-1$, are all in S'. Without loss of generality, we can assume that $\mathrm{names}(\varphi) \cap \mathrm{names}(\psi_i) = \emptyset$, because otherwise the names of the ψ_i's $(i = 2, \ldots, n-1)$ can be moved away via renaming, the resulting pairs of frames will still be in S'. If we can show that $(\varphi\psi_i, \varphi\psi_{i+1}) \in S'$, then transitivity ensures that $(\varphi\varphi_1, \varphi\varphi_2) \in \hat{S}$. Let us now fix i. Then, by the assumption $(\psi_i, \psi_{i+1}) \in S'$, these frames have the form $\psi_i = \psi\{\psi'_1|\ldots|\psi'_m\}$ and $\psi_{i+1} = \psi\{\psi''_1|\ldots|\psi''_m\}$ such that for all $j = 1, \ldots, m$, $(\psi'_j, \psi''_j) \in S$, or $(\psi''_j, \psi'_j) \in S$, or $\psi''_j = \tau_j(\psi'_j)$ for some renaming τ_j, and $\mathrm{names}(\psi) = \emptyset$. If $[\mathrm{names}(\{\psi'_1|\ldots|\psi'_m\}) \setminus \mathrm{names}(\psi_i)] \cap \mathrm{names}(\varphi) \neq \emptyset$, then replace those names with fresh ones in $\{\psi'_1|\ldots|\psi'_m\}$; this can be done, because they don't show up in ψ_i. Similarly for ψ_{i+1}. Let $a_1, \ldots, a_l$ be the names occurring in φ, and let $y_1, \ldots, y_l$ be fresh variables. For $1 \leq k \leq l$, replace every occurrence of a_k in φ by the variable y_k, obtaining a frame ξ such that $\mathrm{names}(\xi) = \emptyset$, and $\varphi\psi_i = (\xi\psi)\{\psi'_1|\ldots|\psi'_m|y_1 = a_1|\ldots|y_l = a_l\}$ and $\varphi\psi_{i+1} = (\xi\psi)\{\psi''_1|\ldots|\psi''_m|y_1 = a_1|\ldots|y_l = a_l\}$. By assumption, $\mathrm{names}(\psi) = \emptyset$, so $\mathrm{names}(\xi\psi) = \emptyset$, and therefore $(\varphi\psi_i, \varphi\psi_{i+1}) \in S'$.

A.4 Proof of Proposition 4

Proof. As a consequence of Proposition 3, it is sufficient to verify that those production rules preserve the computational indistinguishability of the interpretations of frames. For reflexivity, transitivity, and symmetry, this is implied by the fact that computational indistinguishability is an equivalence relation. By the definition of the interpretation of a frame, it is also clear that if ψ is any frame and τ is a renaming, then $[\![\psi]\!]_{\mathfrak{A}} = [\![\tau(\psi)]\!]_{\mathfrak{A}}$.

It is therefore enough to show that if φ_1, φ_2, φ are as in ii of Definition 3, then $[\![\varphi_1]\!]_{\mathfrak{A}} \cong [\![\varphi_2]\!]_{\mathfrak{A}}$ implies $[\![\varphi\varphi_1]\!]_{\mathfrak{A}} \cong [\![\varphi\varphi_2]\!]_{\mathfrak{A}}$. Suppose there is a probabilistic polynomial-time adversary $\mathcal{A}$ whose advantage $|\Pr[\mathcal{A}(\eta, [\![\varphi\varphi_1]\!]_{\mathfrak{A}_\eta}) = 1] - \Pr[\mathcal{A}(\eta, [\![\varphi\varphi_2]\!]_{\mathfrak{A}_\eta}) = 1]|$ is non-negligible in η. This gives an adversary $\mathcal{B}$ that distinguishes φ_1 and φ_2 with non-negligible advantage: Given η and a concrete frame $\widehat{\psi}$ (namely a sample element from either $[\![\varphi_1]\!]_{\mathfrak{A}}$ or $[\![\varphi_2]\!]_{\mathfrak{A}}$), $\mathcal{B}$ simply interprets the frame φ using the values specified by $\widehat{\psi}$ for the variables occurring in φ. All these variables are assigned a unique value if $\widehat{\psi}$ is sampled from $[\![\varphi_i]\!]_{\mathfrak{A}}$ since $\mathrm{var}(\varphi) \subseteq \mathrm{dom}(\varphi_i)$. The adversary $\mathcal{B}$ thus constructs a concrete frame $\widehat{\sigma}_i$, runs $\mathcal{A}(\eta, \widehat{\sigma})$ and outputs the output of $\mathcal{A}$. Since $\widehat{\psi}$ is sampled from $[\![\varphi_i]\!]_{\mathfrak{A}_\eta}$, the distribution of $\widehat{\sigma}_i$ is exactly $[\![\varphi\varphi_i]\!]_{\mathfrak{A}_\eta}$. Therefore the advantage of $\mathcal{B}$, $|\Pr[\mathcal{B}(\eta, [\![\varphi_1]\!]_{\mathfrak{A}_\eta}) = 1] - \Pr[\mathcal{B}(\eta, [\![\varphi_2]\!]_{\mathfrak{A}_\eta}) = 1]|$, equals the advantage of $\mathcal{A}$, which is non-negligible. Furthermore, $\mathcal{B}$ runs in probabilistic polynomial-time since the size of the encoding of φ is constant in η, so each $\widehat{\sigma}_i$ can be computed in probabilistic polynomial time. This proves the claim by contraposition.