On the equality of probabilistic terms*

Gilles Barthe[1], Marion Daubignard[2], Bruce Kapron[3], Yassine Lakhnech[2], and Vincent Laporte[4]

[1] IMDEA Software, Madrid, Spain
[2] VERIMAG, Grenoble, France
[3] University of Victoria, Canada
[4] ENS Cachan, France

Abstract. We consider a mild extension of universal algebra in which terms are built both from deterministic and probabilistic variables, and are interpreted as distributions. We formulate an equational proof system to establish equality between probabilistic terms, show its soundness, and provide heuristics for proving the validity of equations. Moreover, we provide decision procedures for deciding the validity of a system of equations under specific theories that are commonly used in cryptographic proofs, and use concatenation, truncation, and xor. We illustrate the applicability of our formalism in cryptographic proofs, showing how it can be used to prove standard equalities such as optimistic sampling and one-time padding as well as non-trivial equalities for standard schemes such as OAEP.

1 Introduction

Provable security [15] is a methodology used by cryptographers for providing rigorous mathematical proofs of the correctness of cryptographic schemes. One of the popular tools for provable security is the game-based technique [4], in which cryptographic proofs are organized as a sequence of game/event pairs:

$$G_0, A_0 \to^{h_1} G_1, A_1 \to \cdots \to^{h_n} G_n, A_n$$

where G_0, A_0 formalises the security goal—e.g. IND-CPA and IND-CCA for an encryption scheme or UF-CMA and EF-CMA for signature schemes—and the scheme under study, and h_i are monotonic functions such that $\Pr_{G_i}[A_i] \leq h_{i+1}(\Pr_{G_{i+1}}[A_{i+1}])$. By composition, $h_1 \circ \cdots \circ h_n(\Pr_{G_n}[A_n])$ is an upper bound for $\Pr_{G_n}[A_n]$.

While the game-based technique does not advocate any formalism for games, some authors find convenient to model games as probabilistic programs. In this setting, game-based cryptographic proofs often proceed by replacing a set of algebraic expressions $s_1 \ldots s_n$ by another set of expressions $t_1 \ldots t_n$ in the program. The correctness of the transformation is guaranteed provided the tuples of terms $s_1 \ldots s_n$ and $t_1 \ldots t_n$ yield equal distributions. Notable examples include:

* This work was partially supported by French ANR SESUR-012, SCALP, Spanish project TIN2009-14599 DESAFIOS 10, and Madrid Regional project S2009TIC-1465 PROMETIDOS.

One-time padding: for every cyclic group G of prime order and generator g of G, the distributions g^x and $c \cdot g^x$, where the variable x is sampled randomly over $\mathbb{Z}_q$, are equal;

Optimistic sampling: for every k, the distributions $(x, x \oplus y)$ and $(x \oplus y, x)$ are equal, where x is sampled uniformly over the set of bitstrings of size k, and y is an arbitrary but fixed bitstring of size k—here $\oplus$ denotes the bitwise xor on bitstrings.

The purpose of this article is to provide a formalism that captures and justifies the equational reasonings that pervade cryptographic proofs. To this end, we consider an extension of universal algebra that distinguishes between probabilistic variables and determistic variables. While deterministic variables are interpreted in the usual way via valuations, the interpretation of probabilistic variables is through sampling, so that the intepretation $[\![t]\!]_{\boldsymbol{y} \mapsto \boldsymbol{b}}$ of a term t with probabilistic variables $\boldsymbol{x}$ and deterministic variables $\boldsymbol{y}$ under the valuation $\boldsymbol{y} \mapsto \boldsymbol{b}$ is defined as

$$\lambda c \in \sigma.\ \Pr_{\boldsymbol{a} \in \boldsymbol{\tau}}[t[\boldsymbol{x}, \boldsymbol{y} := \boldsymbol{a}, \boldsymbol{b}] = c]$$

where $\boldsymbol{\tau}$ is the type of $\boldsymbol{x}$ and σ is the type of t, and where $\cdot[\cdot := \cdot]$ denotes substitution of variables by values. In the case of optimistic sampling, where the variable x is probabilistic and the variable y is deterministic, the interpretation $[\![x \oplus y]\!]_{y \mapsto b}$ of the expression $x \oplus y$ w.r.t. a valuation $y \mapsto b$ is defined as $a \overset{\$}{\leftarrow} \{0,1\}^k;\ [\![x \oplus y]\!]_{x \mapsto a, y \mapsto b}$, i.e. the distribution obtained by monadic composition of the uniform distribution over $\{0,1\}^k$, and of the (deterministic) interpretation of $\langle\!\langle x \oplus y \rangle\!\rangle_{x \mapsto a, y \mapsto b}$. Equivalently, $[\![x \oplus y]\!]_{y \mapsto b} = \lambda c.\ \Pr_{a \in \{0,1\}^k}[a \oplus b = c]$. Under this interpretation, one can show that

$$[\![\langle x \oplus y, x \rangle]\!]_{y \mapsto b} = \lambda c, d.\ \Pr_{a \in \{0,1\}^k}[a \oplus b = c, a = d]$$

is equal to

$$[\![\langle x, x \oplus y \rangle]\!]_{y \mapsto b} = \lambda c, d.\ \Pr_{a \in \{0,1\}^k}[a = c, a \oplus b = d]$$

Note that the equational theory of probabilistic terms reveals some subtleties: for example, the equation $x \doteq y$ is valid whenever x and y are probabilistic variables of the same type; however, the equation $\langle x, x \rangle \doteq \langle y, y' \rangle$ is not valid in general—as a result, it is important to consider systems of equations rather than single equations, as further explained below.

Our main contributions are:

- the definition of a proof system for reasoning about equations, and systems of equations. We prove that the system is sound and provide useful heuristics for establishing the validity of a system of equations;
- for specific theories, including the theory of xor and concatenation, the definition of decision procedures for deciding the validity of a system of equations; and sufficient conditions for the decidability of the validity of a system of equations.

2 A motivating example

We illustrate the need for proving equality between distributions with one classical example of encryption scheme, namely RSA-OAEP [5, 9]. Recall that an asymmetric encryption scheme is specified by a triple $(\mathcal{KG}, \mathcal{E}, \mathcal{D})$ where $\mathcal{KG}$ is a key generation algorithm which outputs a pair of public and private keys, $\mathcal{E}$ is an encryption algorithm that takes an input a public key and a plaintext algorithm and outputs a ciphertext, and a decryrption algorithm that takes the private key and the ciphertext and produces the corresponding plaintext. An asymmetric encryption scheme $(\mathcal{KG}, \mathcal{E}, \mathcal{D})$ is said to be indistinguishable between real or random (IND-ROR) if the difference between the final distribution of the two games is small:[5]

$$
\begin{array}{l}
(sk, pk) \leftarrow \mathcal{KG}; m \leftarrow A(pk); c \leftarrow \mathcal{E}(pk, m); \mathsf{return}\ c \\
(sk, pk) \leftarrow \mathcal{KG}; m \leftarrow A(pk); y \overset{\$}{\leftarrow} \{0,1\}^k; c \leftarrow \mathcal{E}(pk, y); \mathsf{return}\ c
\end{array}
$$

where A is the procedure that represents the adversary.

OAEP is a famous padding scheme that is used for increasing robustness of RSA encryption. The OAEP algorithm relies on two random oracles G and H, which are sampled during initialization—we gloss over the size of the arguments and images of H and G. Key generation, encryption and decryption are respectively defined as:

$$
\begin{array}{rl}
\mathcal{KG} = & (f, f^{-1}) \overset{\$}{\leftarrow} \Lambda,\ \mathsf{return}\ (f, f^{-1}) \\
\mathcal{E}(m, f) = & r^* \overset{\$}{\leftarrow} \{0,1\}^{k_0};\ s^* \leftarrow (m \mid 0^{k_1}) \oplus G(r^*);\ t^* \leftarrow H(s^*) \oplus r^* \\
& \mathsf{return}\ f(s^* \mid t^*) \\
\mathcal{D}(y) = & s|t := f^{-1}(y); r := H(s) \oplus t; \\
& \mathsf{if}\ [G(r)]_{k_1} = [s]_{k_1}\ \mathsf{then}\ (\mathsf{return}\ [s \oplus G(r)]^{k-k_1})\ \mathsf{else\ reject}
\end{array}
$$

where where Λ denotes the set of trapdoor permutations—for the purpose of this paper, it is sufficient to know that f and f^{-1} are inverse to each other—-and $[.]_k$ and $[.]^k$ respectively denote taking and removing the first k bits of a bitstring.

The first step in the proof of IND-ROR for OAEP is to show that the two code snippets below yield the same distribution:

$$
\begin{array}{ll}
r^* \overset{\$}{\leftarrow} \{0,1\}^{k_0}; m \leftarrow A(f); g^* \overset{\$}{\leftarrow} \{0,1\}^{k-k_0}; & m \leftarrow A(f); y \overset{\$}{\leftarrow} \{0,1\}^k; \\
\mathsf{return}\ f((m \mid 0^{k_1}) \oplus g^* | H((m \mid 0^{k_1}) \oplus g^*) \oplus r^*) & \mathsf{return}\ y
\end{array}
$$

In order to prove the equality, one must show the validity of the equation:

$$f((m \mid 0^{k_1}) \oplus g^* | H((m \mid 0^{k_1}) \oplus g^*) \oplus r^*) \doteq y$$

where g^*, r^*, y are random variables. More formally, one must show that the distribution induced by the left hand side by sampling uniformly g^* and r^* over

[5] Technically, games are indexed by a security parameter η and IND-ROR states that the distance between the families of distributions induced by the indexed games are negligible in η.

their respective sets is the uniform distribution. The informal argument goes as follows: since r^* is uniformly distributed and only occurs once, therefore the expression $H((m \mid 0^{k_1}) \oplus g^*) \oplus r^*$ is uniformly distributed and can be replaced by a fresh random variable hr^*. Thus, we are left to prove

$$f((m \mid 0^{k_1}) \oplus g^* | hr^*) \doteq y$$

Now, g^* is uniformly distributed and only occurs once, therefore the expression $(m \mid 0^{k_1}) \oplus g^*$ is uniformly distributed and can be replaced by a fresh random variable mg^*. Thus, we are left to prove

$$f(mg^* | hr^*) \doteq y$$

The concatenation of random variables being random, one can subsitute $mg^* | hr^*$ by a fresh variable z^*, so that one is left to prove

$$f(z^*) \doteq y$$

To conclude, observe that f is a bijection so $f(z^*)$ is uniformly distributed, and hence we indeed have $f(z^*) \doteq y$. In the course of the paper, we will develop a procedure that formalizes this reasoning.

3 Preliminaries

We refer to e.g. Chapter 8 of [14] for an introduction to finite distributions, with examples from cryptography. Throughout the paper, we only consider (sub)distributions over finite sets: let A be a finite set; the set $\mathcal{D}(A)$ of distributions over A is the set of functions $d : A \to [0,1]$ such that $\sum_{a \in A} d(a) \leq 1$. Given a distribution $d \in \mathcal{D}(A)$ and an element $a \in A$, we write $\mathsf{Pr}[d = a]$ for $d(a)$.

Let A be a finite set of cardinal q. The uniform distribution over A assigns to each element of A probability q^{-1}. We write $x \overset{\$}{\leftarrow} A$ to denote the uniform distribution on A. The monadic composition of the uniform distribution and of a function $f : A \to \mathcal{D}(B)$ is the distribution $y \overset{\$}{\leftarrow} A,\ f(y)$, which is defined by the clause $\mathsf{Pr}[y \overset{\$}{\leftarrow} A,\ f(y) = b] = \frac{q'}{q}$ where q' is the cardinal of $f^{-1}(b)$. Intuitively, this is the distribution of a random variable which is obtained by sampling A uniformly at random to obtain a value y, and then evaluating f at y.

The product distribution $d_1 \times \cdots \times d_n$ of the distributions $d_1 \ldots d_n$ is defined as $x_1 \overset{\$}{\leftarrow} d_1\ \ldots\ x_n \overset{\$}{\leftarrow} d_n,\ (x_1, \ldots, x_n)$. Conversely, the i-th projection of a distribution d over $A_1 \times \cdots \times A_n$ is the distribution $x \overset{\$}{\leftarrow} d,\ \pi_i(x)$, where π_i denotes the usual projection.

The following observation, which only holds for finite domains and uniform distributions, is the cornerstone of the general decision procedure for deciding equality of distributions.

Proposition 1. *For all finite sets A and B, and functions $f, g : A \to B$, the following are equivalent:*

- $x \overset{\$}{\leftarrow} A,\ f(x) = x \overset{\$}{\leftarrow} A,\ g(x)$
- *there exists a bijection $h : A \to A$ such that $f = g \circ h$.*

Note that since A is finite, h is bijective iff it is injective iff it is surjective.

A remark on products Throughout the paper, we use the vector notation to denote tuples of terms. Accordingly, we use tuple notations to denote the product of their types, thus $\boldsymbol{t}$ denotes a tuple of terms and $\boldsymbol{\sigma}$ denotes the product of their types.

4 Syntax and semantics

This section introduces the syntax and semantics of probabilistic terms, and gives a precise formulation of the satisfaction problem for systems of equations of probabilistic terms. For an introduction to equational logic and term rewriting see e.g. [1].

4.1 Syntax

We start from the notion of many-sorted signature. We allow function symbols to be overloaded, but impose restrictions to ensure that terms have at most one sort.

Definition 1 (Signature). *A signature is a triple $\Sigma = (\mathcal{S}, \mathcal{F}, :)$, where $\mathcal{S}$ is a set of sorts, $\mathcal{F}$ is a set of function symbols, and $:$ is a typing relation between function symbols and arities of the form $\sigma_1 \times \ldots \times \sigma_n \to \tau$, with $\sigma_1 \ldots \sigma_n\ \tau \in \mathcal{S}$.*

We require that the typing relation is functional, i.e. if $f : \sigma_1 \times \ldots \times \sigma_n \to \tau$ and $f : \sigma_1 \times \ldots \times \sigma_n \to \tau'$, then $\tau = \tau'$. In particular, we assume that constants have a single type.

Terms are built in the usual way, except that we distinguish between two sets of variables: the set $\mathcal{R}$ denotes variables that are interpreted probabilistically, and the set $\mathcal{D}$ denotes variables that are interpreted deterministically. It is convenient to assume that there are infinitely many deterministic and probabilistic variables of each sort. Moreover, we assume that for every $x \in \mathcal{R}$ there exists a distinguished variable $\bar{x} \in \mathcal{D}$ of the same sort.

Definition 2 (Terms and substitutions). *Let $\Sigma = (\mathcal{S}, \mathcal{F}, :)$ be a signature and let X be a collection of variables. The set $\mathcal{T}_X$ of terms over X is built from the syntax: $t ::= x \mid f(\boldsymbol{t})$ where f ranges over $\mathcal{F}$ and x ranges over X. In the sequel, we consider the set of terms over $\mathcal{V} = \mathcal{D} \cup \mathcal{R}$, and write $\mathcal{T}$ instead of $\mathcal{T}_{\mathcal{D} \cup \mathcal{R}}$. Elements of $\mathcal{T}_{\mathcal{D}}$ are called $\mathcal{D}$-terms.*

Substitutions over X (to $\mathcal{T}_Y$) are defined as functions from X to $\mathcal{T}_Y$; we let $\rho\ t$ denote the result of applying the substitution ρ to t.

Given $Y \subseteq X$, we let $\mathsf{var}_Y(t)$ denote $\mathsf{var}(t) \cap Y$, where $\mathsf{var}(t)$ is defined in the usual way. Moreover, we say that $t \equiv_{\alpha(Y)} t'$ iff there exists a 1-1 renaming $\rho : \mathsf{var}_Y(t) \to \mathsf{var}_Y(t')$ such that $\rho\ t = t'$.

Terms are subject to a simple typing discipline that ensures that functions are applied to arguments of the correct types. In the sequel, we implicitly assume that each variable x has a unique sort σ_x and that terms are well-typed; we adopt the standard notations $t : \sigma$ (resp. $t \in \mathcal{T}_X(\sigma)$) to denote that a term t has type σ (resp. t has type σ and $\mathsf{var}(t) \subseteq X$). Thanks to requiring that typing is functional, every term has at most one type.

Definition 3 (System of equations). *A system of equations over a set X, or X-system of equations, is a statement $s_1 \doteq t_1 \wedge \ldots \wedge s_n \doteq t_n$ where, for $i = 1 \ldots n$, s_i and t_i have the same type, i.e. $s_i, t_i \in \mathcal{T}_X(\sigma_i)$ for some σ_i. We often use $\boldsymbol{s} \doteq \boldsymbol{t}$ as a shorthand for systems of equations.*

Unlike equational logic, it is important to consider systems of equations rather than single equations. Because of the possible dependencies between terms, the conjunction of two valid equalities may not be valid.

Consider the probabilistic variables x, y, z of type σ: the system of equations $x \doteq y \wedge x \doteq z$ is not valid, whereas the two equations $x \doteq y$ and $x \doteq z$ are valid; this is because the distribution $y \overset{\$}{\leftarrow} \sigma, z \overset{\$}{\leftarrow} \sigma,\ \langle y, z\rangle$ yields the uniform distribution over $\sigma \times \sigma$ whereas $x \overset{\$}{\leftarrow} \sigma,\ \langle x, x\rangle$ does not.

Definition 4 (Theory). *A theory is a pair $\mathbb{T} = (\Sigma, E)$ where Σ is a signature and E is a (possibly infinite) set of of systems of equations.*

4.2 Semantics

The semantics of probabilistic terms is adapted immediately from equational logic. In particular, algebras provide the natural semantics for signatures.

Definition 5 (Algebra). *Let $\Sigma = (\mathcal{S}, \mathcal{F}, :)$ be a signature. A Σ-algebra is a pair $\mathbb{A} = ((\mathcal{A}_\sigma)_{\sigma \in \mathcal{S}}, (f_\mathcal{A})_{f \in \mathcal{F}})$ where $\mathcal{A}_\sigma$ is a finite set that interprets the sort σ and $f_\mathcal{A} \in \mathcal{A}_{\sigma_1} \times \cdots \times \mathcal{A}_{\sigma_n} \to \mathcal{A}_\tau$ for every $f \in \mathcal{F}$ such that $f : \sigma_1 \times \ldots \times \sigma_n \to \tau$. In the sequel, we let $\mathcal{A} = \bigcup_{\sigma \in \mathcal{S}} \mathcal{A}_\sigma$ and write $[\![\sigma]\!]$ instead of $\mathcal{A}_\sigma$.*

Terms are interpreted as distributions, by taking a probabilistic interpretation of variables in $\mathcal{R}$.

Definition 6 (Interpretation of terms). *Let $\Sigma = (\mathcal{S}, \mathcal{F}, :)$ be a signature and $\mathbb{A} = ((\mathcal{A}_\sigma)_{\sigma \in \mathcal{S}}, (f_\mathcal{A})_{f \in \mathcal{F}})$ be a Σ-algebra.*

- *An X-valuation is a function $\rho : X \to \mathcal{A}$ such that $\rho(x) \in \mathcal{A}_{\sigma_x}$ for every $x \in X$. We let Val_X denote the set of X-valuations. In the sequel, we often omit the subscript; moreover, we often use the notation $\boldsymbol{x} \mapsto \boldsymbol{a}$ to denote any valuation ρ such that $\rho(x_i) = a_i$.*
- *Let $\rho \in \mathsf{Val}_X$. The pre-interpretation $\langle\!\langle t \rangle\!\rangle_\rho$ of a term $t \in \mathcal{T}_X$ is defined as:*

$$\langle\!\langle t \rangle\!\rangle_\rho = \begin{cases} \rho(t) & \text{if } t \in X \\ f_\mathcal{A}(\langle\!\langle t_1 \rangle\!\rangle_\rho, \ldots, \langle\!\langle t_n \rangle\!\rangle_\rho) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

- *Let $\rho \in \mathsf{Val}_{\mathcal{D}}$. The interpretation $[\![\boldsymbol{t}]\!]_\rho$ of a tuple of terms $\boldsymbol{t}$ of type $\boldsymbol{\sigma}$ is defined by the clause:*

$$[\![\boldsymbol{t}]\!]_\rho = \lambda \boldsymbol{a} : \boldsymbol{\sigma}. \Pr_{\rho' \in \mathsf{Val}_{\mathcal{R}}} [\langle\!\langle \boldsymbol{t} \rangle\!\rangle_{\rho+\rho'} = \boldsymbol{a}]$$

where the valuation $\rho + \rho'$ denotes the (disjoint) union of ρ and ρ', and for every tuple of terms $\boldsymbol{t} = \langle t_1, \ldots, t_n \rangle$ and for every valuation ρ, $\langle\!\langle \boldsymbol{t} \rangle\!\rangle_\rho$ denotes

$$\langle \langle\!\langle t_1 \rangle\!\rangle_\rho, \ldots, \langle\!\langle t_n \rangle\!\rangle_\rho \rangle$$

Note that the interpretation of a tuple of terms needs not coincide with the product distribution of their interpretations. For example, $[\![x, x]\!]_\rho \neq [\![x]\!]_\rho \times [\![x]\!]_\rho$ for every $x \in \mathcal{R}$.

Definition 7 (Model). *Let $\mathbb{T} = (\Sigma, E)$ be a theory; let $\mathbb{A} = ((\mathcal{A}_\sigma)_{\sigma \in \mathcal{S}}, (f_{\mathcal{A}})_{f \in \mathcal{F}})$ be a Σ-algebra.*

- *A system of equations $\boldsymbol{s} \doteq \boldsymbol{t}$ is valid in $\mathbb{A}$, written $\mathbb{A} \models \boldsymbol{s} \doteq \boldsymbol{t}$ iff for every $\rho_{\mathcal{D}} \in \mathsf{Val}_{\mathcal{D}}$, we have $[\![\boldsymbol{s}]\!]_{\rho_{\mathcal{D}}} = [\![\boldsymbol{t}]\!]_{\rho_{\mathcal{D}}}$.*
- *$\mathbb{A}$ is a $\mathbb{T}$-algebra (or $\mathbb{T}$-model) iff for every system of equations $\boldsymbol{s} \doteq \boldsymbol{t} \in E$, we have $\mathbb{A} \models \boldsymbol{s} \doteq \boldsymbol{t}$.*

The notion of model for an equational theory coincides with that of equational logic for theories with $\mathcal{D}$-systems of equations.

Note that one can prove that the following equations are valid: $x \doteq y$ for every probabilistic variables x and y of the same type, $x \oplus x' \doteq y$ for every probabilistic variables x, x' and y of type $\{0, 1\}^k$.

4.3 Satisfaction problem

The problem addressed in this paper can now be stated formally: given a theory $\mathbb{T} = (\Sigma, E)$, a collection of Σ-algebras $(\mathbb{A}_i)_{i \in \mathcal{I}}$ and a system of equations $\boldsymbol{s} \doteq \boldsymbol{t}$, can we decide whether $\forall i \in \mathcal{I}, \mathbb{A}_i \models \boldsymbol{s} \doteq \boldsymbol{t}$. We write $\mathsf{Dec}_{\mathrm{Sat}(\mathbb{T},(\mathbb{A}_i)_{i \in \mathcal{I}})}$ if the problem is decidable.

Stating the satisfaction problem relative to a collection of models rather than a single one is somewhat unusual, and is motivated by the need to carry cryptographic proofs parametrically in the size of the security parameter.

5 Exclusive or, concatenation, and projection

The purpose of this section is to present decision procedures for the theories of exclusive or, and the theory of exclusive or, concatenation, and projection.

$$\begin{array}{rll}
(s \mid t) \oplus (s' \mid t') = & (\downarrow_{(1,\#s')} s \mid (\downarrow_{(\#s'+1,\#s)} s \mid t)) \oplus (s' \mid t') & \text{if } \#s' < \#s \\
(s \mid t) \oplus (s' \mid t') = & (s \mid t) \oplus (\downarrow_{(1,\#s)} s' \mid (\downarrow_{(\#s+1,\#s')} s' \mid t')) & \text{if } \#s < \#s' \\
(s \mid t) \oplus (s' \mid t') = & (s \oplus s') \mid (t \oplus t') & \text{if } \#s = \#s' \\
\downarrow_{(i_1,i_2)} (s \oplus t) = & (\downarrow_{(i_1,i_2)} s) \oplus (\downarrow_{(i_1,i_2)} t) & \\
\downarrow_{(i_1,i_2)} (s \mid t) = & \downarrow_{(i_1,i_2)} s & \text{if } i_2 \leq \#s \\
\downarrow_{(i_1,i_2)} (s \mid t) = & \downarrow_{(i_1-\#s,i_2-\#s)} t & \text{if } \#s < i_1 \\
\downarrow_{(i_1,i_2)} (s \mid t) = & \downarrow_{(i_1,\#s)} s \mid \downarrow_{(1,i_2-\#s)} t & \text{if } i_1 \leq \#s < i_2 \\
\downarrow_{(i_1,i_2)} (\downarrow_{(j_1,j_2)} s) = & \downarrow_{(i_1+j_1,i_2+j_1)} s & \\
\downarrow_{(1,\#s)} (s) = & s &
\end{array}$$

Fig. 1. Theory of concatenation and projection

5.1 Exclusive or

The first theory $\mathbb{T}_\oplus$ has a single sort bs, a constant $0 : \mathsf{bs}$ and a binary symbol $\oplus : \mathsf{bs} \times \mathsf{bs} \to \mathsf{bs}$. Its axioms are:

$$\begin{array}{cc}
x \oplus (y \oplus z) \doteq (x \oplus y) \oplus z & \qquad x \oplus y \doteq y \oplus x \\
x \oplus 0 \doteq x & \qquad x \oplus x \doteq 0
\end{array}$$

We consider the family of algebras $(\mathcal{BS}_k)_{k \in \mathbb{N}}$, where $\mathcal{BS}_k$ is the set of bistrings of size k, with the obvious interpretation for terms. By abuse of notation, we write $\models \boldsymbol{s} \doteq \boldsymbol{t}$ instead of $\forall k \in \mathbb{N},\ \mathcal{BS}_k \models \boldsymbol{s} \doteq \boldsymbol{t}$.

We begin by stating some simple facts. First, one can decide whether $\mathcal{D}$-equations hold.

Lemma 1. *Let $s, t \in \mathcal{T}_\mathcal{D}$. It is decidable whether $\models s \doteq t$.*

Second, one can decide whether a term is semantically equal to a variable in $\mathcal{R}$. We write $\mathcal{U}(t)$ iff for all $\rho \in \mathsf{Val}$, $[\![t]\!]_\rho$ is uniformly distributed.

Lemma 2. *Let $t \in \mathcal{T}$. It is decidable whether $\mathcal{U}(t)$.*

Proof. Every term t can be reduced to a normal form t', in which variables appear at most once. Then $[\![t]\!]_\rho$ is uniformly distributed iff t' contains at least one $\mathcal{R}$-variable.

It follows that one can decide equality of two terms.

Lemma 3. *Let s and t be terms. It is decidable whether $\models s \doteq t$.*

Proof. If $\mathcal{U}(s)$ and $\mathcal{U}(t)$, return true. If $\neg\mathcal{U}(s)$ and $\mathcal{U}(t)$ or $\mathcal{U}(s)$ and $\neg\mathcal{U}(t)$, return false. If $\neg\mathcal{U}(s)$ and $\neg\mathcal{U}(t)$, then s and t can be reduced to normal forms $s' \in \mathcal{T}_\mathcal{D}$ and $t' \in \mathcal{T}_\mathcal{D}$. Return true if $\models s' \doteq t'$ and false otherwise.

In order to extend the result to tuples of terms, we rely on the following lemma. The result is used e.g. in [6].

Proposition 2. *Let $t_1 \ldots t_n \in \mathcal{T}$ such that $\mathcal{U}(t_i)$ for $1 \leq i \leq n$. Exactly one of the following statements hold:*

- $\mathsf{indep}\ (\boldsymbol{t})$*: for every* $\rho \in \mathsf{Val}$, $[\![(t_1 \dots t_n)]\!]_\rho = [\![t_1]\!]_\rho \times \cdots \times [\![t_n]\!]_\rho$*;*
- $\mathsf{dep}\ (\boldsymbol{t})$*: there is a non-null vector* $\boldsymbol{\lambda} \in \{0,1\}^n$ *and* $s \in \mathcal{T}_\mathcal{D}$ *such that* $\mathsf{dep}_{\boldsymbol{\lambda},s}\ (\boldsymbol{t})$*, where* $\mathsf{dep}_{\boldsymbol{\lambda},s}\ (\boldsymbol{t})$ *holds iff for every* $\rho \in \mathsf{Val}$*,*

$$[\![\sum_{1 \leq i \leq n} \lambda_i t_i]\!]_\rho = [\![s]\!]_\rho$$

where $\sum$ *denotes summation* $\bmod 2$.

Proof. For simplicity, assume that $t_1 \dots t_n \in \mathcal{T}_{\{y_1 \dots y_l\}}$ with $y_1 \dots y_l \in \mathcal{R}$, and consider bistrings of length k. Then $\mathsf{indep}\ (\boldsymbol{t})$ iff for all bitstrings of length k $a_1 \dots a_n$, the system of equations

$$(*) \begin{cases} t_1 = a_1 \\ \quad\vdots \\ t_n = a_n \end{cases}$$

has exactly $2^{k(l-n)}$ solutions. Indeed, $\mathsf{Pr}[\bigwedge_{i=1}^n t_i = a_i] = \alpha 2^{-kl}$, where α is the number of solutions of $(*)$. It is now easy to prove by induction on n that $\alpha = 2^{k(l-n)}$ is equivalent to the linear independence of $\boldsymbol{t}$, which is equivalent to $\neg\mathsf{dep}\ (\boldsymbol{t})$.

For example, one can prove that the distribution induced by the triple of terms $(x \oplus y, y \oplus z, z \oplus x)$, where x, y, and z are probabilistic variables of type $\{0,1\}^k$ is not uniformly distributed, i.e. the system of equations:

$$x \oplus y \doteq w_1 \wedge y \oplus z \doteq w_2 \wedge z \oplus x \doteq w_3$$

is not valid, since we have $(x \oplus y) \oplus (y \oplus z) \oplus (z \oplus x) = 0$.

Note that one can effectively decide which of the two conditions hold, since there are only finitely many $\boldsymbol{\lambda}$ to test—and s, if it exists, can be computed from $\sum_{1 \leq i \leq n} \lambda_i t_i$. Decidability follows.

Proposition 3. $\mathsf{Dec}_{\mathrm{Sat}(\mathbb{T}_\oplus, (\mathcal{BS}_k)_{k \in \mathbb{N}})}$.

Proof. The decision procedure works as follows:

1. If the system only contains a single equation $s \doteq t$, invoke Lemma 3;
2. If $\mathsf{indep}\ (\boldsymbol{s})$ and $\mathsf{indep}\ (\boldsymbol{t})$, return true;
3. If $\mathsf{dep}_{\boldsymbol{\lambda},s}\ (\boldsymbol{s})$ and $\mathsf{dep}_{\boldsymbol{\lambda},s}\ (\boldsymbol{t})$ *for the same* $\boldsymbol{\lambda}$ and s, then pick $\lambda_k \neq 0$, and recursively check the smaller system without the equation $s_k \doteq t_k$;
4. otherwise, return false.

Since terms of sort bs are only built from variables of sort bs, decidability extends immediately to the multi-sorted theory $\mathbb{T}_\oplus^+$, with set of sorts bs_k for all k, constants $0^k : \mathsf{bs}_k$, and a—single but overloaded—binary function symbol $\oplus : \mathsf{bs}_i \times \mathsf{bs}_i \to \mathsf{bs}_i$. The axioms are those of $\mathbb{T}_\oplus$. Finally, we consider the algebras $(\mathcal{BS}_k)_{k \in \mathbb{N}}$ with the obvious interpretation.

Proposition 4. $\mathsf{Dec}_{\mathrm{Sat}(\mathbb{T}_\oplus^+, (\mathcal{BS}_k)_{k \in \mathbb{N}})}$.

5.2 Exclusive or, concatenation, and projection

Next, we prove decidability for exclusive or and concatenation. Thus, the theory $\mathbb{T}_{\{\oplus,|,\downarrow\}}$ has infinitely many sorts bs^k and infinitely many function symbols:

$$0^k : \mathsf{bs}^k \qquad \oplus : \mathsf{bs}^k \times \mathsf{bs}^k \to \mathsf{bs}^k \qquad |: \mathsf{bs}^k \times \mathsf{bs}^{k'} \to \mathsf{bs}^{k+k'} \qquad \downarrow_{(i,j)}: \mathsf{bs}^k \to \mathsf{bs}^{j-i+1}$$

where $k', i, j \in \mathbb{N}$ are such that $i \leq j \leq k$. Its axioms are those of the theory $\mathbb{T}_\oplus$, together with axioms for the associativity and neutral for concatenation, and with axioms for relating concatenation, projection and $\oplus$, which are given in Figure 1. Finally, we consider the indexed family of algebras $(\mathcal{BS}_i)_{i\in\mathbb{N}}$ in which the interpretation of bs^k is the set of bitstrings of length ki, with the obvious interpretation of function symbols.

Proposition 5. $\mathsf{Dec}_{\mathrm{Sat}(\mathbb{T}_{\{\oplus,|,\downarrow\}},(\mathcal{BS}_{\leq k})_{k\in\mathbb{N}})}$.

Proof. The proof proceeds by a reduction to the previous case, and relies on a set of rewrite rules that transform an arbitrary system of equations into an equivalent system without concatenation and projection. There are two sets of rewrite rules; both rely on typing information that provides the length of bitstrings; we let $\#s$ denote the length of the bistring s. The first set of rewrite rules, is obtained by orienting the rules of Figure 1 from left to right, and pushes concatenations to the outside and projections to the inside. The second set of rewrite rules, given in Figure 2, aims to eliminate concatenation and projection by transforming equations of the form $s \mid t \doteq s' \mid t'$ with $\#s = \#s'$ and $\#t = \#t'$ into a system of equations $s \doteq s' \mid t \doteq t'$, and by replacing expressions of the form $\downarrow_{(i,j)} x$ by fresh variables $x_{(i,j)}$—in order to get an equivalent system, the replacement is performed by a global substitution $[x := x_{1,i-1} \mid x_{i,j} \mid x_{j+1,\#x}]$.

The procedure terminates: intuitively, the rule for splitting variables can only be applied a finite number of times, and the remaining rules are clearly terminating. Upon termination, one obtains a system of equations of the form $\boldsymbol{s} \doteq \boldsymbol{t} \wedge \boldsymbol{x} \doteq \boldsymbol{u}$ where the ss and ts only contain $\oplus$-terms and the us, are concatenations of variables, and variables on the left hand side, i.e. the xs, do not appear in the first system of equations, and moreover variables arise at most once globally in the us. Thus, the validity of the system is equivalent to the validity of $\boldsymbol{s} \doteq \boldsymbol{t}$ which can be decided by Proposition 3.

6 An equational logic for systems of equations

The purpose of this section is to provide a sound proof system for proving the validity of a system of equations, and to study the conditions under which the proof system is complete and decidable.

$$\begin{array}{rr}
s_1 \mid s_2 \doteq t_1 \mid t_2 \rightarrow \langle \downarrow_{(1,\#t_1)} s_1, (\downarrow_{(\#t_1+1,\#s_1)} s_1) \mid s_2 \rangle \doteq \langle t_1, t_2 \rangle & \text{if } \#t_1 < \#s_1 \\
s_1 \mid s_2 \doteq t_1 \mid t_2 \rightarrow \langle s_1, s_2 \rangle \doteq \langle \downarrow_{(1,\#s_1)} t_1, (\downarrow_{(\#s_1+1,\#t_1)} t_1) \mid t_2 \rangle & \text{if } \#s_1 < \#t_1 \\
s_1 \mid s_2 \doteq t_1 \mid t_2 \rightarrow \langle s_1, s_2 \rangle \doteq \langle t_1, t_2 \rangle & \text{if } \#t_1 = \#s_1 \\
s_1 \mid s_2 \doteq t \oplus t' \rightarrow \langle s_1, s_2 \rangle \doteq \langle \downarrow_{(1,\#s_1)} (t \oplus t'), \downarrow_{(\#s_1+1,\#s_1+\#s_2)} (t \oplus t') \rangle & \\
s_1 \mid s_2 \doteq \downarrow_{(i_1,i_2)} t \rightarrow \langle s_1, s_2 \rangle \doteq \langle \downarrow_{(i_1,i_1+\#s_1)} t, \downarrow_{(i_1+1+\#s_1,i_1+1+\#s_2-\#s_1)} t \rangle &
\end{array}$$

$$\frac{s \doteq t \rightarrow \Delta}{\Gamma \wedge s \doteq t \rightarrow \Gamma \wedge \Delta}$$

$$\Gamma \wedge \downarrow_{(i,j)} x \doteq t \rightarrow (\Gamma \wedge x_{i,j} \doteq t)[x := x_{1,i-1} \mid x_{i,j} \mid x_{j+1,\#x}] \wedge x \doteq x_{1,i-1} \mid x_{i,j} \mid x_{j+1,\#x}$$

Fig. 2. Normalization of equation systems with concatenation and projection

6.1 Proof system

The proof system contains structural rules, equational rules that generalize those of equational logic, and specific rules for probabilistic terms.

Structural rules specifically deal with systems of equations; the rule [Struct] allows us to duplicate, permute, or eliminate equations. Formally $\boldsymbol{s} \doteq \boldsymbol{t} \subseteq \boldsymbol{s'} \doteq \boldsymbol{t'}$ iff for every j there exists i such that the i-th equation of $\boldsymbol{s'} \doteq \boldsymbol{t'}$ is syntactically equal to the j-th equation of $\boldsymbol{s} \doteq \boldsymbol{t}$. Moreover, the rule [Merge] allows us to merge systems of equations, provided they do not share any variables in $\mathcal{R}$. Note that the side condition of the [Merge] rule is necessary for soundness; without the side condition, one could derive for probabilistic variables x, y, z of the same type that $x \doteq y \wedge x \doteq z$ is valid (since from $x \doteq y$ and $x \doteq z$ are), which is unsound as mentioned earlier.

The equational rules include reflexivity, symmetry and transitivity of equality, congruence rules for function symbols, a rule for axioms, and a substitution rule. Note that the rule for functions is stated for ensuring soundness, and that the following rule is unsound:

$$\frac{\vdash s_1 \doteq t_1 \ldots \vdash s_n \doteq t_n}{\vdash f(s_1 \ldots s_n) \doteq f(t_1 \ldots t_n)}$$

because it would allow to derive that $\vdash x \oplus x \doteq y \oplus z$ for x, y, z probabilistic variables of type $\{0,1\}^k$. Note also that we allow in the application of the [Fun] rule to have side equations $\boldsymbol{u} \doteq \boldsymbol{v}$, which is required to have successive applications of the [Fun] rule.

Likewise, the rule for substitutions requires that the subsituted terms are deterministic; without this restriction, the rule would be unsound as for every deterministic variable y of type $\{0,1\}^k$ and probabilistic variable x of the same type, one could derive $\vdash x \doteq x \oplus y[y := x]$ from $\vdash x \doteq x \oplus y$. Note that one can combine the rule for substitution with the rule [Rand] below to allow substitutions of terms that contain *fresh* probabilistic variables, in the style of [11].

Finally, the rules for probabilistic variables include rules for α-conversion, and the rule [Bij], that is the syntactical counterpart of Proposition 1. It assumes that

$\mathsf{var}_{\mathcal{R}}(\boldsymbol{s})\cup\mathsf{var}_{\mathcal{R}}(\boldsymbol{t}) \subseteq \boldsymbol{x}$, and requires that there are $\mathcal{D}$-terms $\boldsymbol{u}$ and $\boldsymbol{v}$ that represent bijections, and such that the composition of $\boldsymbol{u}$ with $\boldsymbol{s}$ is equal to $\boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}]$—where $\bar{\boldsymbol{x}} \in \mathcal{D}$ is a type-preserving renaming of $\boldsymbol{x}$. In the side condition, we let V_R denote $\mathsf{var}_{\mathcal{R}}(\boldsymbol{s}) \cup \mathsf{var}_{\mathcal{R}}(\boldsymbol{t})$ and V_D denote $\mathsf{var}_{\mathcal{D}}(\boldsymbol{s}) \cup \mathsf{var}_{\mathcal{D}}(\boldsymbol{t})$.

$$\frac{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}{\vdash \boldsymbol{s'} \doteq \boldsymbol{t'}}\text{[Struct] where } \boldsymbol{s'} \doteq \boldsymbol{t'} \subseteq \boldsymbol{s} \doteq \boldsymbol{t} \qquad \frac{\boldsymbol{s} \doteq \boldsymbol{t} \subseteq E}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}\text{[Axm]}$$

$$\frac{\vdash \boldsymbol{s} \doteq \boldsymbol{t} \qquad \vdash \boldsymbol{s'} \doteq \boldsymbol{t'}}{\vdash \boldsymbol{s} \doteq \boldsymbol{t} \wedge \boldsymbol{s'} \doteq \boldsymbol{t'}}\text{[Merge] where } (\mathsf{var}_{\mathcal{R}}(\boldsymbol{s}) \cup \mathsf{var}_{\mathcal{R}}(\boldsymbol{t})) \cap (\mathsf{var}_{\mathcal{R}}(\boldsymbol{s'}) \cup \mathsf{var}_{\mathcal{R}}(\boldsymbol{t'})) = \emptyset$$

$$\frac{}{\vdash \boldsymbol{s} \doteq \boldsymbol{s}}\text{[Refl]} \qquad \frac{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}{\vdash \boldsymbol{t} \doteq \boldsymbol{s}}\text{[Sym]} \qquad \frac{\vdash \boldsymbol{s} \doteq \boldsymbol{t} \qquad \vdash \boldsymbol{t} \doteq \boldsymbol{u}}{\vdash \boldsymbol{s} \doteq \boldsymbol{u}}\text{[Trans]}$$

$$\frac{\vdash \boldsymbol{u} \doteq \boldsymbol{v} \wedge s_1 \doteq t_1 \wedge \ldots \wedge s_n \doteq t_n}{\vdash \boldsymbol{u} \doteq \boldsymbol{v} \wedge f(s_1 \ldots s_n) \doteq f(t_1 \ldots t_n)}\text{[Fun]}$$

$$\frac{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}{\vdash \rho\boldsymbol{s} \doteq \rho\boldsymbol{t}}\text{[Subst]where } \rho : \mathcal{D} \to \mathcal{T}_{\mathcal{D}}$$

$$\frac{\boldsymbol{s} \equiv_{\alpha(\mathcal{R})} \boldsymbol{s'} \qquad \boldsymbol{t} \equiv_{\alpha(\mathcal{R})} \boldsymbol{t'} \qquad \vdash \boldsymbol{s} \doteq \boldsymbol{t}}{\vdash \boldsymbol{s'} \doteq \boldsymbol{t'}}\text{[Alpha]}$$

$$\frac{\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}] \qquad \vdash \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}] \doteq \bar{\boldsymbol{x}}}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}\text{[Bij]where } \begin{cases} V_R \subseteq \boldsymbol{x} \\ V_D \cap \bar{\boldsymbol{x}} = \emptyset \\ (\mathsf{var}(\boldsymbol{u}) \cup \mathsf{var}(\boldsymbol{v})) \subseteq (\bar{\boldsymbol{x}} \cup V_D) \end{cases}$$

Fig. 3. Proof system

Here is an example of the use of this system to prove optimistic sampling, i.e. for every deterministic variable y of type $\{0,1\}^k$ and probabilistic variable x of the same type, $\vdash x \oplus y \doteq x$. The last step of the proof is an application of the [Bij] rule, with $u = \bar{x} \oplus y$ and $v = \bar{x} \oplus y$. It is easy to check that the premises hold, i.e. $\vdash x \oplus y[x := \bar{x} \oplus y] \doteq y$, and $\vdash \bar{x} \oplus y[\bar{x} := \bar{x} \oplus y] \doteq \bar{x}$.

One application of the [Bij] rule is to lift equality of deterministic terms to equality of distributions. Concretely, we have:

$$\frac{\vdash \boldsymbol{s}[\boldsymbol{x} := \bar{\boldsymbol{x}}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}]}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}\text{[Rand]where } \begin{cases} V_R \subseteq \boldsymbol{x} \\ V_D \cap \bar{\boldsymbol{x}} = \emptyset \end{cases}$$

Using this rule, one can also conclude that for every distinct probabilistic variable x and y of type $\{0,1\}^k$, one has $\vdash x \oplus y \doteq x$.

In order to apply optimistic sampling in context, we must rely on a derived rule for linear variables. Given a tuple of terms $\boldsymbol{s}$ in which x of type σ appears exactly once, and assuming that $\vdash x \doteq t$ with $\mathsf{var}(t) \cup \mathsf{var}(\boldsymbol{s}) \subseteq \boldsymbol{y}$, then:

$$\frac{x \doteq t \wedge \boldsymbol{y} \doteq \boldsymbol{y}}{\vdash \boldsymbol{s} \doteq \boldsymbol{s}[x := t]}\text{[Linear] } x \ \not\in \mathsf{var}(\boldsymbol{t})$$

The rule [Linear] can be proved by induction on the structure of the terms, or using the [Bij+] rule in the next section. In particular, one can prove that for every theory that contains the $\oplus$ operator and its associated equations that:

$$\frac{}{\vdash \boldsymbol{s} \doteq \boldsymbol{s}[x := x \oplus t]}\ x \ \notin \mathsf{var}(t) \wedge x \text{ linear in } \boldsymbol{s}$$

Note that the conjunct $\boldsymbol{y} \doteq \boldsymbol{y}$ is required in the rule [Linear] because one could otherwise take s to be $x \oplus y$ and t to be y, to prove $(x \oplus y)[x := y] \doteq x$, which is of course not valid.

6.2 Soundness

The proof system is sound.

Proposition 6. *Let $\mathbb{T} = (\Sigma, E)$ be a theory and assume that $\vdash \boldsymbol{s} \doteq \boldsymbol{t}$. For every $\mathbb{T}$-algebra $\mathbb{A}$, we have $\mathbb{A} \models \boldsymbol{s} \doteq \boldsymbol{t}$.*

Proof (Sketch). By induction on the length of derivations. We only consider the case [Bij]. Assume that we have $\mathbb{A} \models \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}]$ and $\mathbb{A} \models \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}] \doteq \bar{\boldsymbol{x}}$. To show that $\mathbb{A} \models \boldsymbol{s} \doteq \boldsymbol{t}$, i.e. $[\![\boldsymbol{s}]\!]_\rho = [\![\boldsymbol{t}]\!]_\rho$ for every valuation $\rho \in \mathsf{Val}_{\mathcal{D}}$. We have (the second equality holds by induction hypothesis):

$$[\![\boldsymbol{s}]\!]_{\boldsymbol{x} \mapsto [\![\boldsymbol{u}]\!]_\rho} = [\![\boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}]]\!]_\rho = [\![\boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}]]\!]_\rho$$

To conclude, it is sufficient to show that for every $\boldsymbol{a} \in [\![\sigma_{\boldsymbol{x}}]\!]$, and partial valuation ρ' with domain $(\mathsf{var}(\boldsymbol{u}) \cup \mathsf{var}(\boldsymbol{v})) \setminus \boldsymbol{x}$ the function $[\![\boldsymbol{u}]\!]_{\boldsymbol{\rho'} + \bar{\boldsymbol{x}} \mapsto \boldsymbol{a}}$ is a bijection from $[\![\sigma_{\boldsymbol{x}}]\!]$ to itself. By induction hypothesis, we have that
$[\![\boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}]]\!]_{\rho' + \bar{\boldsymbol{x}} \mapsto \boldsymbol{a}} = (\rho' + \bar{\boldsymbol{x}} \mapsto \boldsymbol{a})\bar{\boldsymbol{x}}$, or equivalently $[\![u]\!]_{\rho' + \bar{\boldsymbol{x}} \mapsto [\![\boldsymbol{v}]\!]_{\bar{\boldsymbol{x}} \mapsto \boldsymbol{a}}} = \boldsymbol{a}$. Hence $[\![\boldsymbol{u}]\!]_{\boldsymbol{\rho'} + \bar{\boldsymbol{x}} \rightarrow \boldsymbol{a}}$ is a bijection.

6.3 Products

Our proof system does not make any specific provision with product, thus it is not possible to prove that for every probabilistic variables x, y and z of respective types $\{0,1\}^k$, $\{0,1\}^{k'}$ and $\{0,1\}^{k+k'}$ one has $\vdash x|y = z$. Thus, the proof system is incomplete.

One can remedy to this issue by considering theories with products, and enriching the proof system for such theories.

Definition 8 (Theory with products). *A theory $\mathbb{T} = (\Sigma, E)$ has products iff for every sorts σ and σ', there exists a sort τ and function symbols $\pi : \tau \to \sigma$, $\pi' : \tau \to \sigma'$ and $\mathsf{pair} : \sigma \times \sigma' \to \tau$ such that the following $\mathcal{D}$-equations hold:*

$$\mathsf{pair}(\pi(y), \pi'(y)) \doteq y \qquad \pi(\mathsf{pair}(x, x')) \doteq x \qquad \pi'(\mathsf{pair}(x, x')) \doteq x'$$

Concatenation and truncation of bitstrings are the primary examples of function symbols that yield a product structure. Given a theory with products, one can show that the rules for products are sound:

$$\frac{\vdash \boldsymbol{s}[x, x' := \pi(y), \pi'(y)] \doteq \boldsymbol{t}}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}\text{[ProdE]}$$

The [ProdE] rule implicitly assumes that products exist, and that y is a fresh variable. The rule allows to collate two probabilistic variables x and x' of respective sorts σ and σ' by a probabilistic variable y of sort $\sigma \times \sigma'$, and is useful to prove the previous example. There is a dual rule [ProdI], which allows to introduce projections, and is ommitted.

6.4 Example revisited

The example of Section 2 can be established through successive applications of the [Linear] rule, the [Prod] rule, and finally the [Bij] rule. The signature is that of bitstrings with exclusive or, concatenation, and truncation, extended with two function symbols f and f^{-1}, with additional axioms that state that f and f^{-1} are mutually inverse bijections.

The first step in the derivation is to show that the equation $f(z^*) \doteq y$ is derivable, for y and z^* probabilistic variables. The equation is established using the [Bij] rule, and relies on the axioms on f and f^{-1}. Formally, we prove:

$$f(z^*) \doteq y$$

Then, the second step in the proof is to derive from the above equality the equation:

$$f(mg^* \mid hr^*) \doteq y$$

The proof proceeds as follows: we use the [Prod] rule to establish that $mg^* \mid hr^* \doteq z^*$, and then the [Fun] rule to prove that $f(mg^* \mid hr^*) \doteq f(z^*)$, so by transitivity, we have $f(mg^* \mid hr^*) \doteq y$. Then, we can apply the [Linear] rule to conclude that

$$f((m \mid 0^{k_1}) \oplus g^* | hr^*) \doteq y$$

By a further application of the [Linear] rule, one concludes as expected that:

$$f((m \mid 0^{k_1}) \oplus g^* | H((m \mid 0^{k_1}) \oplus g^*) \oplus r^*) \doteq y$$

6.5 Towards completeness

The purpose of this section is to define completeness, and to provide some partial results towards completeness. Unfortunately, we have not been able to prove completeness for any theory of interest.

Recall that a proof system is complete w.r.t. a set of models if all systems of equations that are valid in the models are also provable.

Definition 9 (Completeness). *Let $\mathbb{T} = (\Sigma, E)$ be a theory. The proof system is complete (resp. $\mathcal{D}$-complete) wrt an indexed family $(\mathbb{A}_i)_{i\in\mathcal{I}}$ of $\mathbb{T}$-algebras iff for every system of equations (resp. $\mathcal{D}$-equations) $\boldsymbol{s} \doteq \boldsymbol{t}$, if for all $i \in \mathcal{I}$, $\mathbb{A}_i \models \boldsymbol{s} \doteq \boldsymbol{t}$ then $\vdash \boldsymbol{s} \doteq \boldsymbol{t}$.*

There are two main issues with completeness. The first issue is the existence of products, which is discussed above. The second and main difficulty is the representation of bijections in the syntax. Indeed, one must show that the rule [Bij] does indeed provide a syntactic counterpart to Proposition 1. In other words, completeness requires that one can represent some bijections by a tuple of terms, so that the rule [Bij] applies. A stronger hypothesis, namely that all functions are representable by terms, is captured by the definition of primal algebra, which is used e.g. in [13]: an algebra $\mathbb{A}$ is primal iff for every function $f : \sigma_1 \times \ldots \times \sigma_n \to \tau$ (with $n > 0$, and $\sigma_1 \ldots \sigma_n, \tau$ interpretations of sorts) there exists a $\mathcal{D}$-term u with free variables $x_1 : \sigma_1 \ \ldots \ x_n : \sigma_n$ such that for every $(a_1, \ldots, a_n) \in \sigma_1 \times \cdots \times \sigma_n$, we have:

$$[\![u]\!]_{(x_1:=a_1,\ldots,x_n:=a_n)} = f(a_1, \ldots, a_n)$$

Unfortunately, the notion of primal algebra is too strong for our setting, because proving completeness would require that *all* the algebras of the indexed family $(\mathbb{A}_i)_{i\in\mathcal{I}}$ are primal. Since the size of the algebras is unbounded, it is not clear, even for the case of bitstrings considered in Section 5, how to define the signature so to meet this requirement. One can instead consider a weaker notion, called weak primality.

Definition 10 (Weakly primal). *An algebra $\mathbb{A}$ is weakly primal iff for every $f_1, f_2 : \sigma_1 \times \ldots \times \sigma_n \to \tau$ (with $n > 0$, and $\sigma_1 \ldots \sigma_n, \tau$ interpretations of sorts) that are interpretations of $\mathcal{D}$-terms, and for every bijection $h : \sigma_1 \times \ldots \times \sigma_n \to \sigma_1 \times \ldots \times \sigma_n$ such that $f_2 = f_1 \circ h$, there exist terms $u_1, \ldots, u_n$ with free variables $x_1, \ldots, x_n$ such that $f_2 = f_1 \circ [\![(u_1, \ldots, u_n)]\!]$, and $[\![(u_1, \ldots, u_n)]\!]$ is a bijection over $\sigma_1 \times \ldots \times \sigma_n$.*

Note that weak primality does not require that h is representable, but instead that there exist terms that satisfy the same equation as h. This weakening of the original definition is necessary to prove that weak primality holds for the signature of $\oplus$. The proof uses similar arguments to the proof of decidability of validity of equations, and yields a process to build the terms $u_1 \ldots u_n$. We illustrate the process on two examples: assume that $s = x_1 \oplus x_2 \oplus x_3$ and $t = x_2 \oplus x_3$. Then one takes the terms $u_1 = x_1$, $u_2 = x_1 \oplus x_2$, $u_3 = x_3$, which provide a bijection. Now, assume that $s = x_1 \oplus x_2 \oplus x_3$ and $t = x_3$. Then one takes the terms $u_1 = x_1$, $u_2 = x_2$, $u_3 = x_3 \oplus x_1 \oplus x_2$, which provide a bijection.

Weak primality is sufficient to prove that every valid equation (*not* system of equations) is derivable, provided that completeness holds for every $\mathcal{D}$-equation. The idea of the proof is as follows. Consider an equation $s \doteq t$ with deterministic variables $x_1 \ldots x_n$ of type $\sigma_1 \ldots \sigma_n$, and probabilistic variables $y_1 \ldots y_m$ of type $\tau_1 \ldots \tau_m$. Assume that for all $i \in \mathcal{I}$, $\mathbb{A}_i \models \boldsymbol{s} \doteq \boldsymbol{t}$. By Proposition 1, there exists a bijection $f_{a_1 \ldots a_n} : \llbracket \tau_1 \rrbracket \times \ldots \times \llbracket \tau_m \rrbracket \to \llbracket \tau_1 \rrbracket \times \ldots \times \llbracket \tau_m \rrbracket$ for every $(a_1 \ldots a_n) \in \llbracket \sigma_1 \rrbracket \times \cdots \times \llbracket \sigma_n \rrbracket$ such that:

$$\langle\!\langle \boldsymbol{s} \rangle\!\rangle_{\boldsymbol{x} \mapsto \boldsymbol{a}, \boldsymbol{y} \mapsto \boldsymbol{b}} = \langle\!\langle \boldsymbol{t} \rangle\!\rangle_{\boldsymbol{x} \mapsto \boldsymbol{a}, \boldsymbol{y} \mapsto f_{a_1 \ldots a_n}(\boldsymbol{b})}$$

for every $(b_1, \ldots, b_m) \in \llbracket \tau \rrbracket$. By weak primality, there exist $\mathcal{D}$-terms $\boldsymbol{u}$ and $\boldsymbol{v}$ with free variables $x_1 \ldots x_n$ and $\bar{y_1} \ldots \bar{y_m}$ such that for every $a_1 \ldots a_n$ $b_1 \ldots b_m$, we have:

- $f_{a_1 \ldots a_n}(b_1, \ldots, b_m) = \langle\!\langle \boldsymbol{u} \rangle\!\rangle_{\boldsymbol{x} \mapsto \boldsymbol{a}, \bar{\boldsymbol{y}} \mapsto \boldsymbol{b}}$,
- $f^{-1}_{a_1 \ldots a_n}(b_1, \ldots, b_m) = \langle\!\langle \boldsymbol{v} \rangle\!\rangle_{\boldsymbol{x} \mapsto \boldsymbol{a}, \bar{y} \mapsto b}$.

By $\mathcal{D}$-completeness, we have that $\vdash \boldsymbol{s} \doteq \boldsymbol{t}[\boldsymbol{y} := \boldsymbol{u}]$ and $\vdash \boldsymbol{u}[\bar{\boldsymbol{y}} := \boldsymbol{v}] \doteq \bar{\boldsymbol{y}}$, and hence by applying the [Bij] rule it follows that $\vdash \boldsymbol{s} \doteq \boldsymbol{t}$ is provable.

6.6 Proof automation

One strategy for proving a system of equations is to apply the [Bij] rule so as to fall back in the realm of universal algebra—i.e. of $\mathcal{D}$-systems of equations. Thus, applicability of the [Bij] rule is the key to automation. This section considers conditions for automating the [Bij] rule. Our starting point is a mild generalization of the [Bij] rule, which does not require that all probabilistic variables are eliminated simulateneously (recall that V_D is a shorthand for $\mathsf{var}_{\mathcal{D}}(\boldsymbol{s}) \cup \mathsf{var}_{\mathcal{D}}(\boldsymbol{t})$):

$$\frac{\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}] \qquad \vdash \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}] \doteq \bar{\boldsymbol{x}}}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}\,[\text{Bij+}]$$

where $V_D \cap \bar{\boldsymbol{x}} = \emptyset$ and $\mathsf{var}(\boldsymbol{u}) \cup \mathsf{var}(\boldsymbol{v}) \subseteq (\bar{\boldsymbol{x}} \cup V_D)$. The difference with the rule [Bij] is that the side condition $\mathsf{var}_{\mathcal{R}}(\boldsymbol{s}) \cup \mathsf{var}_{\mathcal{R}}(\boldsymbol{t}) \subseteq \boldsymbol{x}$ is dropped. Informally, this rule allows constructing the underlying bijection of Proposition 1 incrementally. It does not increase the power of the logic, but makes it easier to use, and is important for injectivity, as suggested below. There is a close connection between the rule [Bij+] and matching, see e.g. [1].

Definition 11 (1-1 matching problem). *Let s, t be two terms and $\boldsymbol{x} \subseteq \mathsf{var}_{\mathcal{R}}(s) \cup \mathsf{var}_{\mathcal{R}}(t)$. A solution to a 1-1 matching problem is a pair $(\boldsymbol{u}, \boldsymbol{v})$ of $\mathcal{D}$-terms such that:*

- $\mathsf{var}_{\mathcal{D}}(\boldsymbol{u}) \cup \mathsf{var}_{\mathcal{D}}(\boldsymbol{v}) \subseteq \bar{\boldsymbol{x}}$,
- $\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}]$,
- $\vdash \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}] \doteq \bar{\boldsymbol{x}}$.

We let $\mathcal{S}ol(\boldsymbol{s} \mathrel{-\!\!\prec\!\!\!\prec}^{1-1}_{\boldsymbol{x}} \boldsymbol{t})$ denote the set of solutions.

The rule [Bij+] can be rephrased equivalently as:

$$\frac{\mathcal{S}ol(\boldsymbol{s} \mathrel{-\!\!\prec\!\!\!\prec}^{1-1}_{\boldsymbol{x}} \boldsymbol{t}) \neq \emptyset}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}$$

Hence, for every system of equations $\boldsymbol{s} \doteq \boldsymbol{t}$, we have that $\mathcal{S}ol(\boldsymbol{s} \prec\!\!\prec^{1-1}_{\boldsymbol{x}} \boldsymbol{t}) \neq \emptyset$ implies that for every $i \in \mathcal{I}$, we have $\mathbb{A}_i \models \boldsymbol{s} \doteq \boldsymbol{t}$. Thus, one can prove a system of equations $\boldsymbol{s} \doteq \boldsymbol{t}$ by exhibiting an element of $\mathcal{S}ol(\boldsymbol{s} \prec\!\!\prec^{1-1}_{\boldsymbol{x}} \boldsymbol{t})$.

Call a tuple of $\mathcal{D}$-terms $\boldsymbol{s}$ injective (w.r.t. variables $\boldsymbol{x}$ and theory E) iff for every $\boldsymbol{e}$, $\vdash \boldsymbol{s} \doteq \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{e}]$ implies $\vdash \boldsymbol{e} \doteq \boldsymbol{x}$. Note that every vector of terms is injective whenever E has unitary matching. Moreover, for every single variable x and expression s in the theory of bitstrings, s is injective w.r.t. x, provided x is provably equal to $x \oplus s_0$, and x does not occur in s_0. On the other hand, one cannot prove injectivity for terms that contain two variables x and y: indeed, let s be $x \oplus y$. Then for every constant bitstring c one can derive $x \oplus y \doteq x \oplus y[x, y := x \oplus c, y \oplus c]$ whereas we do not have $x \doteq x \oplus c$ and $y \doteq y \oplus c$. This explains why it is important to consider the rule [Bij+] instead of [Bij].

The rule [Match] below, that uses the notion of injective term as a side condition, is derivable:

$$\frac{\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}] \qquad \vdash \boldsymbol{s}[\boldsymbol{x} := \bar{\boldsymbol{x}}] \doteq \boldsymbol{t}[\boldsymbol{x} := \boldsymbol{v}]}{\vdash \boldsymbol{s} \doteq \boldsymbol{t}}\text{[Match]if } \boldsymbol{s} \text{ injective w.r.t. } \boldsymbol{x}$$

Assume that $\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}] \doteq \boldsymbol{t}[\boldsymbol{x} := \bar{\boldsymbol{x}}]$ and $\vdash \boldsymbol{s}[\boldsymbol{x} := \bar{\boldsymbol{x}}] \doteq \boldsymbol{t}[\boldsymbol{x} := \boldsymbol{v}]$. Then, $\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}][\bar{\boldsymbol{x}} := \boldsymbol{v}] \doteq \boldsymbol{t}[\bar{\boldsymbol{x}} := \boldsymbol{v}]$ by substitution. That is, $\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}]] \doteq \boldsymbol{t}[\bar{\boldsymbol{x}} := \boldsymbol{v}]$. By transitivity, $\vdash \boldsymbol{s}[\boldsymbol{x} := \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}]] \doteq \boldsymbol{s}$ and by injectivity $\vdash \boldsymbol{u}[\bar{\boldsymbol{x}} := \boldsymbol{v}] \doteq \bar{\boldsymbol{x}}$. One concludes by applying the rule [Bij+].

Thus, one can automate proofs in our logic by performing matching on injective terms.

7 Conclusion

We have considered a mild extension of universal algebra in which variables are given a probabilistic interpretation. We have given a sound proof system and useful heuristics to carry equational reasoning between such probabilistic terms; moreover, we have provided decision procedures for specific theories that arise commonly in cryptographic proofs.

Related work Equational logic [10] is a well-established research field, and there has been substantial work to develop proof systems and decision procedures for differents flavours of the logic: many-sorted, multi-sorted, etc. Yet there seems to have been few works that consider probabilistic extensions of equational logic; for example, P-Maude [12] is an extension of Maude that supports probabilistic rewrite theories, an extension of term rewriting where a term rewrites to another term with a given probability. However, none of these works seems to have been motivated by cryptography.

Equational theories have been thoroughly studied in the setting of cryptographic protocols; see e.g. [7]. In particular, computational and probabilisitc semantics for an equational theory of exclusive or is given, in the context of a more general approach to such semantics for general equational theories, is given in [3]. However, this work does not consider equational logics with probabilistic terms.

Future work The formalism of probabilistic terms seems new and deserves further investigation in its own right. It would be interesting to develop further the proof theory of probabilistic terms, and in particular to establish sufficient conditions for completeness and decidability. In addition, it seems relevant to study its relationship with other probabilistic extensions of equational logic, such as P-Maude [12]. The connection between matching and 1-1 matching also deserves further attention.

Our work is part of a larger effort to carry a proof-theoretical study of logical methods for cryptographic proofs, and our main focus will be to exploit our results in cryptography. Further work is required to extend the scope of our results to other theories of interest for cryptography, see e.g. [7], including permutations, exponentiation, etc. We also intend to extend our results to (loop-free) probabilistic programs, and to develop automated proof methods to decide observational equivalence between such programs. A further step would be to consider, instead of observational equivalence, a notion of statistical distance between programs and to develop automated approximation methods.

In the long term, our goal is to implement our methods and integrate them in tools to reason about cryptographic schemes and protocols, e.g. CertiCrypt [2], or our automated tool to reason about encryption [8].

Acknowledgements The authors are grateful to the anonymous reviewers for their comments, and to Santiago Escobar, Pascal Lafourcade, and José Meseguer for interesting discussions.

References

1. Franz Baader and Tobias Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.
2. Gilles Barthe, Benjamin Grégoire, and Santiago Zanella Béguelin. Formal certification of code-based cryptographic proofs. In *Proceedings of POPL'09*, pages 90–101. ACM Press, 2009.
3. Mathieu Baudet, Véronique Cortier, and Steve Kremer. Computationally sound implementations of equational theories against passive adversaries. *Inf. Comput.*, 207(4):496–520, 2009.
4. M. Bellare and P. Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In *Advances in Cryptology – EUROCRYPT'06*, volume 4004 of *Lecture Notes in Computer Science*, pages 409–426. Springer-Verlag, 2006.
5. Mihir Bellare and Philipp Rogaway. Optimal asymmetric encryption – How to encrypt with RSA. In *Advances in Cryptology – EUROCRYPT'94*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111. Springer-Verlag, 1995.
6. Emmanuel Bresson, Yassine Lakhnech, Laurent Mazaré, and Bogdan Warinschi. A generalization of ddh with applications to protocol analysis and computational soundness. In Alfred Menezes, editor, *CRYPTO*, volume 4622 of *Lecture Notes in Computer Science*, pages 482–499. Springer, 2007.
7. Véronique Cortier, Stéphanie Delaune, and Pascal Lafourcade. A survey of algebraic properties used in cryptographic protocols. *Journal of Computer Security*, 14(1):1–43, 2006.

8. Judicaël Courant, Marion Daubignard, Cristian Ene, Pascal Lafourcade, and Yassine Lakhnech. Towards automated proofs for asymmetric encryption schemes in the random oracle model. In *Proceedings of CCS'08*, pages 371–380. ACM Press, 2008.
9. E. Fujisaki, T. Okamoto, D. Pointcheval, and J. Stern. RSA-OAEP is secure under the RSA assumption. *Journal of Cryptology*, 17(2):81–104, 2004.
10. J. A. Goguen and J. Meseguer. Completeness of many-sorted equational logic. *SIGPLAN Not.*, 16(7):24–32, 1981.
11. Russell Impagliazzo and Bruce M. Kapron. Logics for reasoning about cryptographic constructions. *Journal of Computer and Systems Sciences*, 72(2):286–320, 2006.
12. Nirman Kumar, Koushik Sen, José Meseguer, and Gul Agha. A rewriting based model for probabilistic distributed object systems. In Elie Najm, Uwe Nestmann, and Perdita Stevens, editors, *FMOODS*, volume 2884 of *Lecture Notes in Computer Science*, pages 32–46. Springer, 2003.
13. Tobias Nipkow. Unification in primal algebras, their powers and their varieties. *J. ACM*, 37(4):742–776, 1990.
14. V. Shoup. *A Computational Introduction to Number Theory and Algebra.* Cambridge University Press, 2008.
15. Jacques Stern. Why provable security matters? In *Advances in Cryptology – EUROCRYPT'03*, volume 2656 of *Lecture Notes in Computer Science*, pages 449–461. Springer-Verlag, 2003.